Software Design for a 38.4 kbaud Data Terminal

INTRODUCTION

This Application Note describes a CRT terminal designed around the DP8350 CRT controller and the INS8080 microprocessor. The hardware is a modified version of the circuit described in Application Note AN-199. The software was redesigned and optimized for terminal speed and function. In its present form it is upwards compatible with the Hazeltine 1500 video terminal and has a limited graphics capability. Furthermore, it is able to communicate with a host computer via an RS-232 port, at 38.4 kbaud, without using fill-in characters or handshaking. One 2k by 8 EPROM contains all the software required to implement the terminal. An optional EPROM can be used to add features such as menu display or to transform the terminal into a calculator (in the local mode). The absence of the second EPROM does not affect the operation of the terminal as the software checks for its presence.

DATA TERMINAL FEATURES

- Modes: remote/local
- Limited graphics
- Window scrolling
- Line transmitting and local editing
- Hazeltine 1500 compatible*
- Video display: two pages, 24 x 80 characters/page
- Upper/lower case
- Scrolling plus screen roll up/roll down
- Cursor: blinking (two rates)
- Line, character insert/delete
- Attributes: dual intensity/inverse video
- Full duplex RS-232 port; 110-38400 baud
- Keyboard input: 7-bit parallel
- Full cursor control and addressing
- Cursor enable/disable
- Single board (BLC/SBC) compatible design

LEAD

IN

*The majority of the software written for the Hazeltine 1500 will run with no modification. However, there are differences.

**Includes the ASCII characters A-Q, a-q, space and DEL.

GRAPHICS

ON

National Semiconductor Application Note 270 Wong Hee Nick Samaras February 1982



UNIQUE FEATURES

Graphics Capability: The graphics capability of this terminal, although limited by the number of symbols (34), proves to be very helpful. Typical uses include digital waveform generation (e.g., logic analyzer display), and graph oriented displays such as histograms. A graphics menu is available in the local mode. Entering $\uparrow Q \uparrow$ from the keyboard will result in a two line menu display. Line 23 displays upper and lower case characters, while line 24 displays the corresponding graphics symbols (see *Figure 3*). In local, entering $\uparrow B$ will switch the terminal to the graphics mode; the ESC key can be used to exit. In remote mode, the format requirements for graphics display generation are summarized by the flow-chart shown at the bottom of this page.

The same flowchart can be used in local, if the "lead-in": block is omitted.

Typical transmission sequences are:

- 7E, 02, 42, 10, 1B
- 7E, 02, 63, 10, 10, 10, ... ,10, 1B

7E, 02, 42, 8, 8, 8, 4A, 7E, 0C, 7E, 0C, 1B

All the graphics symbols, along with the upper and lower case characters, are coded into one 2716 EPROM. As a result, both the character set and the graphics symbols may be customized. The total number of available fonts is 128. The field on each displayed character is 7 rows by 10 columns. The alphanumeric symbols occupy a 5 by 7 subfield typically, except for those requiring descenders; they occupy a 5 by 9 section, while the graphics symbols utilize the whole 7 by 10 field.

Transmit: The data terminal can transmit one line of text upon receipt of the 14H code from the keyboard in local mode. Alternately, the host CPU can request transmission by sending 14H prefixed by the 7E lead-in code.

Note that ↑ indicates a control key entry. Lead-in code: 7E.

CURSOR MOVE

UP, DOWN,

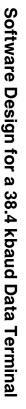
LEFT. RIGHT.

CHARACTER INPUT

GRAPHICS

SYMBOL

SELECT*





© 1995 National Semiconductor Corporation TL/F/5869

RRD-B30M115/Printed in U. S. A

FSC/

CLEAR

SCREEN

The same function can be used in a relatively unconventional way when programming in BASIC. The majority of BASIC interpreters used in small business systems or home computers incorporate a line-oriented editor, almost adequate for most of the tasks they have to perform. The basic problem with such editors is that they cannot change the flow of the program easily. In other words they cannot change line numbers. This is a shortcoming, as it is both annoying and tedious having to retype segments of text in order to change the program flow, just because the editor cannot handle altering line numbers only.

This terminal offers an efficient solution to this problem. Simply stated, it allows changing line numbers only. Here is a brief description of a typical sequence leading to text and/ or line number modification. Let us assume that a BASIC interpreter is used and that the program that needs to be changed is in memory. Using the list command, the program lines to be modified can be displayed. Now, while in the Command Mode of BASIC, the terminal is switched to local. The user has effectively at his disposal a screen-oriented editor. The cursor can be moved about and text changed as desired; that, of course, includes line numbers. When the editing is completed, the user positions the cursor on the line that was altered and types T. In response, the cursor scans the line, inverting the attributes. At the same time the line is transmitted to the host CPU in the same order as it was scanned, from left to right. Attribute inversion serves as feedback to the user. After the last character of each line has been transmitted, the cursor returns to the beginning of the following line. As a result, consecutive T keyboard entries transmit successive lines. Thus, altering the flow of a BASIC program involves entering the local mode, changing line numbers, transmitting the modified program lines, and switching back to on-line operation. All this can be accomplished at a fraction of the time usually required otherwise. Finally, entering similar lines of text such as the ones found in "PRINT" statements, can be accomplished easily by switching to local, typing the first line and transmitting it; then moving the cursor up one line, changing the line number along with parts of the text that are different, retransmitting the line, and so on. In this way the user can create a long program segment while operating repetitively on one line.

Insert/Delete with Range: This is a rather unusual function that can assist in generating pseudo "screen window" effects. Specifically, a pre-selected number of display lines can scroll while the rest of the display remains fixed. Each "window" is defined as N lines by 80 characters, where: $1 \le N \le 48$, counting from the current cursor location to the end of page. The brief BASIC program that follows demonstrates the use of this function. In this example the display lines 1 through 4, and 19 through 24 remain "frozen". The message (100 lines long) is displayed on lines 5 through 18, demonstrating the scrolling of a section of the display.

```
100 PRINT CHR$(&H7E) + CHR$(&H11) + "d";
110 FOR I = 1 TO 100
120 PRINT CHR$(&H7E) + CHR$(&H1D)
+ CHR$(&H49) + CHR$(12);
130 PRINT, "WINDOW SCROLLING LINE:",I,
CHR$(&H0D)
```

```
140 NEXT I
```

80 Character Software FIFO: This is one of the key items that allows terminal communication at 38.4 kbaud without handshaking. An 80 character first-in, first-out software buffer is used. The incoming characters are stored temporarily in this buffer, while the microprocessor is servicing interrupts. As time becomes available, the characters are retrieved from the FIFO and processed. That includes performing a terminal function or moving an ASCII character to the video memory. The software allows for a large number of concurrent service requests such as row start, keyboard, as well as multiple ACE interrupts.

Fast Service Routine for Row Start Interrupt: Conventional row start address look-up and loading are not done during the row start interrupt time; instead, a simple row counting routine is used. The terminal count (a software counter) generates a triggering signal for video RAM wraparound address loading. The use of this technique improves the system throughput substantially. Cursor and Top of the Page address loading (i.e., writing to the appropriate DP8350's registers) is done during the vertical retrace interval.

Keyboard Controlled Mode Selection: The operating mode of the terminal can be selected from the keyboard. To aid the user in identifying which mode the terminal is in, two cursor blinking rates are used. The low rate indicates remote mode; a high rate indicates local.

Other functions that can be selected from the keyboard are:

- 1) Upper/lower case. The default mode upon power up is determined by reading the SW3 switch setting.
- 2) Next page. A software switch that selects for display page one or two.

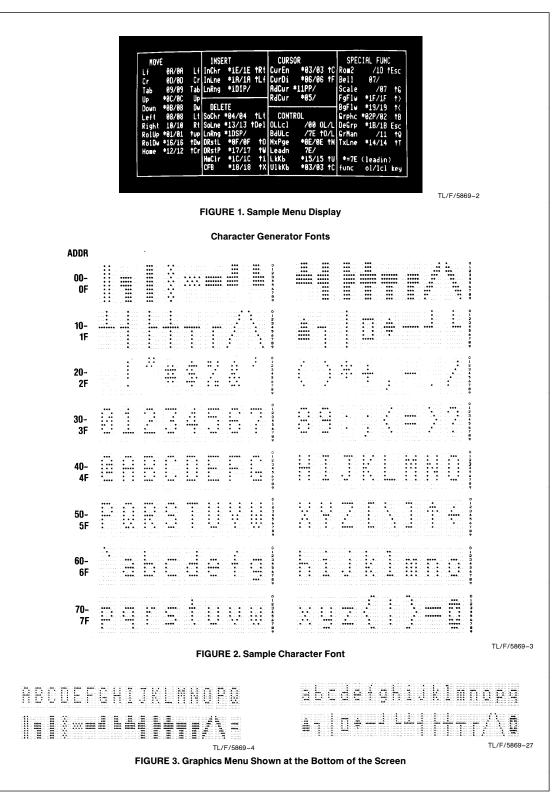
Read Cursor: In the local mode the present cursor location can be displayed on line 24, columns 79–80. For example, if the cursor is located on line 8, column 66, entering \uparrow E from the keyboard will result in a display of "Ag" at the bottom right hand corner of the screen. This can save time in looking up the ASCII equivalent codes of the X, Y cursor coordinates to be used in cursor addressing. (Note that, \uparrow E = ENQ = 05H.)

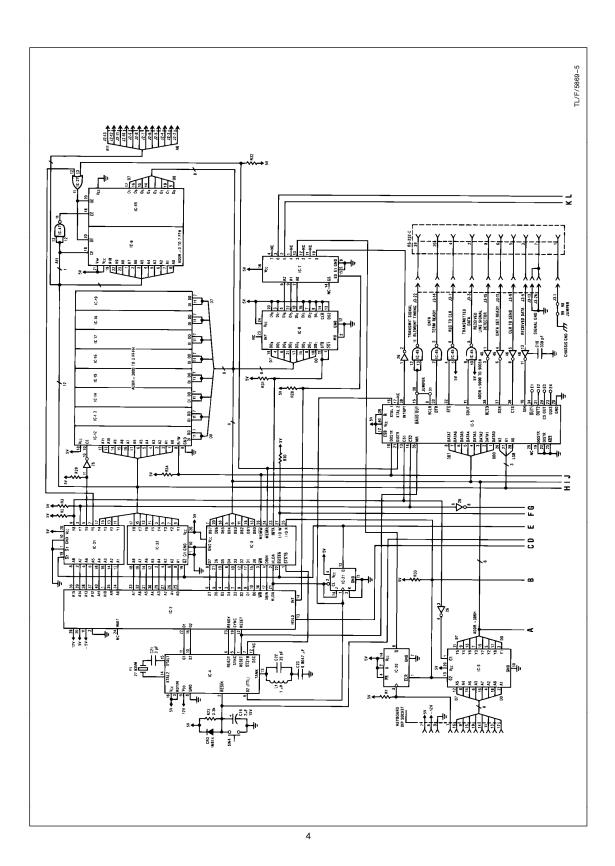
The following is an example of how this could be used in a BASIC program.

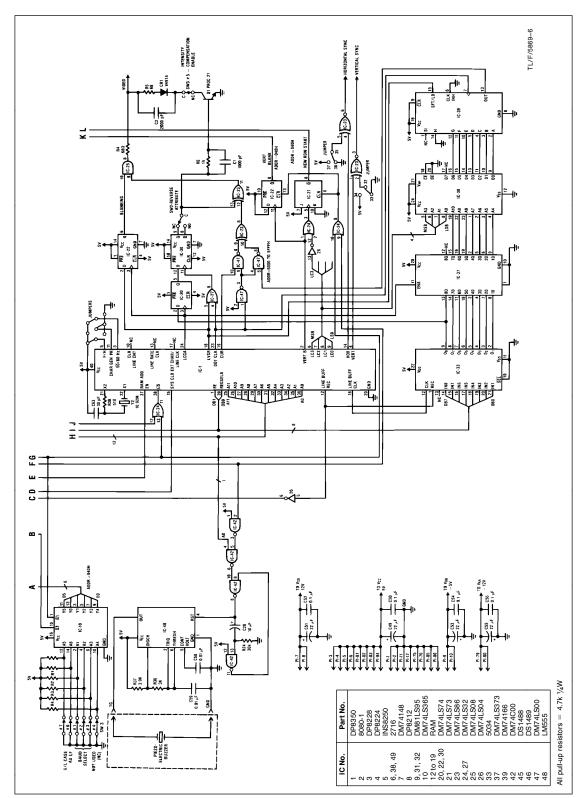
PRINT CHR\$ (&H7E) + CHR\$ (&H11) + "Ag"

Upon execution of the above statement, the cursor will move to line 8, column 66.

Menu Display: In the local mode the user has access to a menu display that summarizes the terminal's functions, along with the corresponding control codes (see *Figure 1*). This feature is optional and resides in EPROM #2. The important thing to note is that various kinds of menu/HELP displays can be implemented easily in this fashion. This function can be accessed from the keyboard. Alternately, a dedicated HELP key (that generates the 1D code) can be used.









Cursor Move/Control Line Feed 0A / 0A Sarriage Return 0D / 0D Tab 03 / 09 Dursor Up 7E, 0C / 0C Cursor Down 7E, 06 / 0B Cursor Light 10 / 10 Home and Clear 7E, 12 / 12 Home and Clear 7E, 12 / 12 Home and Clear 7E, 16 / 10 Disble Cursor 7E, 06 / 06 Address Cursor 7E, 14 / 14 Line Insert 7E, 10 / 10 Dharacter Insert 7E, 10 / 9, Y Remote Only Person Participation Particip	Functions	On-Line / Local	rol Functions Summary Remarks
Line Feed 0A / 0A Carriage Return 0D / 0D Tab 09 / 09 Cursor Up 7E, 0C / 0C Cursor Down 7E, 0B / 0B Cursor Left 0B / 0B Cursor Left 10 / 10 Home and Clear 7E, 12 / 12 Home and Clear 7E, 16 / 10 Enable Cursor 7E, 05 / 05 Insert 7E, 14 / 14 Line Insert 7E, 12 / 12 Character Insert 7E, 12 / 14 Line Insert 7E, 12 / 12 Character Strip 7E, 04 / 04 Character Delete 7F / 7F Line Delete 7E, 13 / 13 Line Delete 7E, 13 / 13 Line Delete 7E, 17 / 17 Miscellaneous Local /Remote 1 / 10 Clear to End of Line 7E, 15 / 15 Keyboard Lock 7E, 15 / 15 Septial Functions Function Menu 1 / 10 Graphics Off 7E, 18 / 18 Septial Functions Function Menu 1 / 10 Graphics Off 7E, 18 / 14 Character Strip 7E, 02 / 02 Enter End of Page 7E, 17 / 17 Miscellaneous Local /Remote 1 / 00 Local Only Special Functions Function Menu 1 / 10 Graphics Off 7E, 18 / 18 Caphics Off 7E, 18 / 18 Caphics Off 7E, 18 / 19 Caphics Off 7E, 18 / 19 Caraptics Off 7E, 19 / 19 Clear Foreground Follows 7E, 19 / 19 Clear Foreground Follows 7E, 19 / 19 Clear Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Cack 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Cack 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Cack 7E, 19 / 19 Clear Foreground 7E, 19 / 19 Clear Foreground 7E, 19 / 19 Clear Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Clear 7D (Clear 7E, 18 / 18 Clear 7D (Cle			Tomarks
Carriage Return 00 / 0D Tab 00 / 0D Tab 00 / 0D Cursor Up 7.E, 06 / 0C Cursor Down 7.E, 06 / 0C Cursor Left 06 / 0B Cursor Fight 10 / 10 Home and Clear 7.E, 12 / 12 Home and Clear 7.E, 12 / 12 Home and Clear 7.E, 16 / 1C Enable Cursor 7.E, 06 / 06 Address Cursor 7.E, 10 / 10 Delable Cursor 7.E, 10 / 10 Line Insert 7.E, 11 / X / Y Remote Only Remote Only Delate Cursor 7.E, 04 / 04 Character Strip 7.E, 10 / 39, Y Character Delete 7.E, 11 / X / Y Character Delete 7.E, 10 / 31 / 31 Line Delete with Range 7.E, 10 / 31 / 31 Line Delete with Range 7.E, 10 / 31 / 31 Local Only Local Only Upper/Lower Case / 7.E Keyboard Unick 7.E, 02 / 02 Keyboard Unick 7.E, 02 / 03 Graphics On 7.E	Cursor Move/Control		
Tab 09 / 09 Cursor Up 7E, 06 / 06 Cursor Left 08 / 08 Cursor Hight 10 / 10 Home 7E, 12 / 12 Home and Clear 7E, 10 / 10 Enable Cursor 7E, 06 / 06 Address Cursor 7E, 11, X, Y Read Cursor 7E, 11, X, Y Remote Only Remote Only Read Cursor 7E, 10, 49, Y Remote Only Remote Only Delete 7E, 04 / 04 Character Strip 7E, 04 / 04 Character Strip 7E, 04 / 04 Character Strip 7E, 04 / 04 Character Delete 7F / 7F Line Delete 7E, 10, 53, Y Clear to End of Page 7E, 06 / 06 Local Only Local Only Upper/Lower Case / 7E Local Nemote / 00 Local Only Local Only	Line Feed	0A / 0A	
Cursor Up 7E, 06 / 06 Cursor Left 08 / 08 Cursor Fight 10 / 10 Home 7E, 10 / 12 Home and Clear 7E, 10 / 10 Home and Clear 7E, 10 / 10 Enable Cursor 7E, 03 / 03 Disable Cursor 7E, 05 / 05 Address Cursor 7E, 11, X, Y / Read Cursor 7E, 10 / 10 Ine Insert 7E, 11, X, Y / Character Insert 7E, 12 / 12 Line Insert 7E, 11, X, Y / Character Insert 7E, 10 / 10, 49, Y Enemote Only Delete Character Strip 7E, 04 / 04 Character Strip 7E, 07 / 07 Clear to End of Line 7E, 07 / 07 Clear to End of Line 7E, 07 / 07 Clear to End of Page 7E, 07 / 07 Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 02 / 02 Entor graphics Mode. Graphics On 7E, 02 / 02 Entor graphics mode. Graphics On 7E, 10 / 10 A summary of available functions and thei	Carriage Return	0D / 0D	
Cursor Down 7E, 08 / 08 Cursor Left 08 / 08 Cursor Left 08 / 08 Cursor Left 08 / 08 Cursor Light 10 / 10 Home and Clear 7E, 12 / 12 Home and Clear 7E, 03 / 03 Disable Cursor 7E, 05 / 05 Insert Character Insert 7E, 12 / 12 Character Insert 7E, 11, X, Y Remote Only Address Cursor 7E, 11, X, Y Remote Only Delete Character Insert 7E, 10, 49, Y Remote Only Delete Character Delete 7F / 7F The Control Control Character Delete 7E, 10, 53, Y Remote Only Cocal Only Clear to End of Line 7E, 07 / 0F Local Only Local Only Clear to End of Line 7E, 06 / 0F Local Only Local Only Dipper/Lower Case / 7E Local Only Local Only Dipper/Lower Case / 7E Local Only Local Only Next Page 7E, 02 / 02 Enter graphics mode. Cocal Only	Tab	09 / 09	
Cursor Left 08 / 08 Cursor Night 10 / 10 Home 7E, 12 / 12 Home and Clear 7E, 10 / 10 Enable Cursor 7E, 03 / 03 Disable Cursor 7E, 03 / 03 Disable Cursor 7E, 05 / 05 Address Cursor 7E, 11, X, Y / Read Cursor 7E, 11, X, Y / Read Cursor 7E, 11, X, Y / Read Cursor 7E, 11, X, Y / Remote Only Remote Only Insert 7E, 12, 14, 17A Character Strip 7E, 04 / 04 Character Strip 7E, 05, 97 Character Strip 7E, 10, 79, 77 Character Delete 7E, 13 / 13 Line Delete with Range 7E, 07, 07 Clear to End of Line 7E, 06 / 05 Clear to End of Page 7E, 07, 07 Keyboard Lock 7E, 03 / 03 Beil 07 / Miscellaneous 1 Local Only Local Only Keyboard Lock 7E, 03 / 03 Beil 07 / <t< td=""><td>Cursor Up</td><td>7E, 0C / 0C</td><td></td></t<>	Cursor Up	7E, 0C / 0C	
Cursor Right 10 / 10 home and Clear 7E, 12 / 12 home and Clear 7E, 12 / 10 Enable Cursor 7E, 03 / 03 Disable Cursor 7E, 05 / 05 Address Cursor 7E, 05 / 05 Insert Character Insert 7E, 17, 14, 74 Line Insert 7E, 17, 14, 74 Line Insert 7E, 17, 14, 74 Line Insert 7E, 10, 49, Y Remote Only Character Strip Character Strip 7E, 04 / 04 Character Delete 7F, 7F Line Delete 7E, 13 / 13 Line Delete 7E, 10, 53, Y Clear to End of Page 7E, 17 / 17 Miscellaneous Local Only Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Wast Page 7E, 02 / 02 Enter Only Special Functions 7 Remote Only Special Functions 7 Remote Only Graphics On 7E, 02 / 02 Enter graphics mode. Graphics On	Cursor Down	7E, 0B / 0B	
Home TE, 12 / 12 Home and Clear TE, 1C / 1C Enable Cursor TE, 03 / 03 Disable Cursor TE, 06 / 06 Address Cursor TE, 11, X, Y / Read Cursor TE, 15, X, Y / Remote Only Delete Character Insert TE, 15, 14, 14 Line Insert With Range TE, 10, 49, Y Remote Only Delete Character Delete TE, 13, 13 Line Delete with Range TE, 15, 5, Y / Clear to End of Line TE, 15, 7, Y Remote Only Clear to End of Line TE, 15, 5, Y / Remote Only Clear to End of Page TE, 10, 53, Y / Remote Only Clear to End of Line TE, 15, 5, Y / Remote Only Clear to End of Page TE, 15, 5, Y / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On TE, 02 / 02 Enter graphics mode. Graphics On TE, 10, 20, 20 Graphics On TE, 10, 21, 22 Enter graphics mode. Graphics On TE, 102 / 02 Enter graphics mode. Graphics On TE, 11 / 11 Enter Stansmit TE, 14 / 14 Foreground Follows TE, 19 / 19 Clear Foreground Te, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1-80). This is an aid for graphics and text alignment (local).	Cursor Left	08 / 08	
Home and Clear TE, 1C / 1C Enable Cursor 7E, 03 / 03 Disable Cursor 7E, 05 / 05 Read Cursor 7E, 11, X, Y Remote Only Read Cursor 7E, 11, X, Y Remote Only Address Cursor 7E, 12 / 1E Line Insert 7E, 12 / 12 Line Insert 7E, 12 / 14 Line Insert 7E, 12 / 14 Line Insert 7E, 10, 49, Y Remote Only Delete Character Strip 7E, 04 / 04 Character Delete 7E, 13 / 13 Line Delete 7E, 13 / 13 Line Delete 7E, 13 / 13 Line Delete 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Upper/Lower Case / 7E Local Only Special Functions Function Menu / 1D Graphics On 7E, 02 / 02 Enter graphics mode. Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 Evit graphics mode. Graphics Menu / 11 Line Transmit 7E, 14 / 14 Foreground Follows 7E, 17 / 17 Remote Only Character Strip 7E, 02 / 02 Enter graphics mode. Graphics Menu / 11 Line Za Jighlay upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Scale / 07 Remote Only Cursor Strip 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1-80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Cursor Right	10 / 10	
Enable Cursor 7E, 03 / 03 Deable Cursor 7E, 06 / 06 Address Cursor 7E, 11, X/ Remote Only Read Cursor 7E, 11, X/ Remote Only TE, 10, 49, Y Remote Only Delete Character Insert 7E, 12 / 14 Line Insert With Range 7E, 10, 49, Y Remote Only Delete Character Strip 7E, 04 / 04 Character Delete 7F, 17, 13 Line Delete With Range 7E, 10, 53, Y Remote Only Character Delete 7F, 13 / 13 Line Delete With Range 7E, 10, 53, Y Remote Only Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 15 / 15 Keyboard Lock 7E, 13 / 13 Ball 07 / Remote Only Special Functions Function Menu 1 D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Menu 1 Line 24 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Scaler Orly 7E, 19 / 19 Scaler Orly 7E, 19 / 19 Scaler Orly 7E, 19 / 19 Scaler Orgonum 7E, 19 / 19 Scaler Orgonum 7E, 19 / 19 Scaler Orgonum 7E, 19 / 10 Scale (07) 7E, 10 / 10 Scale	Home	7E, 12 / 12	
Disable Cursor 7E, 11, X, Y / Remote Only Read Cursor 7E, 11, X, Y / Remote Only Insert 7E, 12 / 12 Character Insert 7E, 12 / 12 Line Insert 7E, 1A / 1A Line Delete 7E, 1D, 49, Y Remote Only Delete 7E, 13 / 13 Line Delete 7E, 10, 53, Y / Clear to End of Line 7E, DF / OF Clear to End of Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Next Page 7E, 0E / 0E Veyboard Lock 7E, 05 / 0B Bell 07 / Remote Only Special Functions Function Menu / 1D Graphics On 7E, 12 / 13 Graphics On 7E, 12 / 14 Graphics Menu / 11 Line Transmit 7E, 14 / 14 Foreground Follows 7E, 17 / 19 Graphics Menu / 11 Line Transmit 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Cara Foreground 7E, 18 / 18 Scale / 07 Function II 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Cara Foreground 7E, 18 / 18 Scale / 07 Function Vey 7E, 01 / 01	Home and Clear	7E, 1C / 1C	
Address Cursor 7E, 11, X, Y / Remote Only Read Cursor 7E, 12, X, Y / Remote Only Insert Character Insert 7E, 12 / 12 Line Insert 7E, 12 / 14 Line Delete 7E, 13, 73 Line Delete 7E, 15, 73 / 7 Line Delete with Range 7E, 10, 53, Y / Remote Only Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 17, 17 Miscellaneous Local/Remote / 00 Local Only Next Page 7E, 0F / 0E Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 State 7F, 19 / 19 Line Transmit 7E, 14 / 14 Foreground Follows 7E, 17 / 19 Char Teground Follows 7E, 17 / 19 Remote Only Clear to End of Page 7E, 17 / 17 Miscellaneous Local Only A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 Exit graphics mode. Graphics Off 7E, 18 / 18 Exit Corresponding graphics symbols (local). Transmits the cursor line and inverts its attributes. The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local).	Enable Cursor	7E, 03 / 03	
Read Cursor 7E, 05 / 05 Insert 7E, 1E / 1E Line Insert 7E, 1A / 1A Line Insert 7E, 1D, 49, Y Remote Only Delete Character Strip 7E, 10, 49, Y Character Strip 7E, 10, 49, Y Character Delete 7F, 77 Line Delete with Range 7E, 10, 53, Y / Clear to End of Line 7E, 10, 53, Y / Clear to End of Page 7E, 17, 17 Miscellaneous /00 Local Only Local Only Upper/Lower Case /7E Keyboard Lock 7E, 15, 15 Keyboard Lock 7E, 15, 15 Keyboard Lock 7E, 16, 70 Remote Only Enter graphics mode. Graphics On 7E, 02 / 02 Graphics Off 7E, 18 / 18 Graphics Off 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Clear Foreground Follows 7E, 18 / 18 Scale /07 Ine Transmit 7E, 14 / 14 Foreground Follows 7E, 18 / 18 Scale /07	Disable Cursor	7E, 06 / 06	
Insert 7E, 1E / 1E Character Insert 7E, 1A / 1A Line Insert 7E, 1D, 49, Y Remote Only Delete Character Strip 7E, 04 / 04 Character Delete 7E, 13 / 13 Line Delete with Range 7E, 10, 53, Y / Character Delete 7E, 17 / 17 Miscellaneous /200 Local Ange 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 18 / 18 Transmits the cursor line and inverts its attributes. Graphics Menu 71 Line 23 displays upper and lower case	Address Cursor	7E, 11, X, Y /	Remote Only
Character Insert 7E, 1E / 1E Line Insert 7E, 1A / 1A Line Insert 7E, 1D, 49, Y Remote Only Delete Character Strip 7E, 04 / 04 Character Delet 7F / 7F Line Delete 7E, 13 / 13 Line Delete 7E, 17 / 17 Miscellaneous Local Or Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Char Foreground 7E, 18 / 18 Scale / 07 Function Wenu 7E, 01 / 01	Read Cursor	7E, 05 / 05	
Line Insert TE, 1A / 1A Line Insert with Range TE, 1D, 49, Y Pelete Character Strip TE, 04 / 04 Character Delete TF / 7F Line Delete TE, 13 / 13 Line Delete TE, 13 / 13 Line Delete TE, 10, 53, Y / Clear to End of Line TE, 0F / 0F Clear to End of Page TE, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Local Only Local Only Local Only Local Only Local Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On TE, 02 / 02 Enter graphics mode. Graphics Off TE, 18 / 18 Exit graphics mode. Graphics Off TE, 18 / 18 Exit graphics mode. Graphics Menu / 11 Line Transmit TE, 14 / 14 Foreground Follows TE, 19 / 19 Clear Foreground Te, 18 / 18 Scale / 07 Fol Up 7E, 01 / 01 Fol Up 7E, 01 / 01	Insert		
Line Insert with Range 7E, 1D, 49, Y Remote Only Delete Character Strip 7E, 04 / 04 Character Delete 7E, 13 / 13 Line Delete 7E, 13 / 13 Line Delete 7E, 10, 53, Y Remote Only Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 1D, 53, Y Local Only Miscellaneous Local/Remote / 00 Local Only Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 13 / 15 Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 Exit graphics mode. Graphics Off 7E, 14 / 14 Foreground Follows 7E, 17 / 17 Remote Only Clear Foreground 7E, 18 / 18 Scale / 07 Remote Only The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Character Insert	7E, 1E / 1E	
Delete Character Strip 7E, 04 / 04 Character Delete 7F / 7F Line Delete 7E, 13 / 13 Line Delete with Range 7E, 10, 53, Y / Dice to End of Line 7E, 07 / 0F Clear to End of Line 7E, 17 / 17 Miscellaneous / 00 Local Only Local/Remote / 00 Local Only Vay Page 7E, 05 / 0F Local Only Next Page 7E, 06 / 0E Local Only Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 Transmits the cursor line and inverts its attributes. Foreground Fo	Line Insert	7E, 1A / 1A	
Character Strip 7E, 04 / 04 Character Delete 7F / 7F Line Delete 7E, 13 / 13 Line Delete with Range 7E, 1D, 53, Y / Remote Only Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Seell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Line Insert with Range	7E, 1D, 49, Y	Remote Only
Character Delete 7F / 7F Line Delete 7E, 13 / 13 Line Delete with Range 7E, 10, 53, Y / Remote Only Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock Keyboard Lock 7E, 15 / 15 Keyboard Unlock Special Functions 7E Local mode only Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 1F / 1F Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 The line above the cursor becomes a scale (1–80). This is a	Delete		
Line Delete 7E, 13 / 13 Line Delete with Range 7E, 1D, 53, Y / Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 Clear Foreground Follows 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Character Strip	7E, 04 / 04	
Line Delete with Range 7E, 1D, 53, Y / Remote Only Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 18 / 18 Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Character Delete	7F / 7F	
Clear to End of Line 7E, 0F / 0F Clear to End of Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 Clear Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Line Delete	7E, 13 / 13	
Clear to End of Page 7E, 17 / 17 Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 Clear Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Line Delete with Range	7E, 1D, 53, Y /	Remote Only
Miscellaneous Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Menu Ine Transmit 7E, 18 / 18 Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 17 / 19 Teansmits the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local).	Clear to End of Line	7E, 0F / 0F	
Local/Remote / 00 Local Only Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Clear to End of Page	7E, 17 / 17	
Upper/Lower Case / 7E Local Only Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Graphics On 7E, 18 / 18 Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Foreground Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Miscellaneous		
Next Page 7E, 0E / 0E Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions 7E, 02 / 02 Enter graphics mode. Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 Clear Foreground Clear Foreground 7E, 18 / 18 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 Texperies and text alignment (local).	Local/Remote	/ 00	Local Only
Keyboard Lock 7E, 15 / 15 Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 Texpendence	Upper/Lower Case	/ 7E	Local Only
Keyboard Unlock 7E, 03 / 03 Bell 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 Texpendence	Next Page	7E, 0E / 0E	
Beil 07 / Remote Only Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 19 / 19 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 Text of the correspondence of the correspondence of the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local).	Keyboard Lock	7E, 15 / 15	
Special Functions Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F Background Follows 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Keyboard Unlock	7E, 03 / 03	
Function Menu / 1D A summary of available functions and their corresponding codes (local mode only). Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F Background Follows 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Bell	07 /	Remote Only
Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F Background Follows 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Special Functions		
Graphics On 7E, 02 / 02 Enter graphics mode. Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F Background Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Function Menu	/ 1D	
Graphics Off 7E, 1B / 1B Exit graphics mode. Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F Transmits the cursor line and inverts its attributes. Background Follows 7E, 19 / 19 Tel ine above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 Tel ine above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local).	Graphics On	7E. 02 / 02	
Graphics Menu / 11 Line 23 displays upper and lower case characters and line 24 the corresponding graphics symbols (local). Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F Transmits the cursor line and inverts its attributes. Background Follows 7E, 19 / 19 Test is a state of the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01 7E, 01 / 01	Graphics Off		
Line Transmit 7E, 14 / 14 Transmits the cursor line and inverts its attributes. Foreground Follows 7E, 1F / 1F Foreground Follows Background Follows 7E, 19 / 19 Foreground Clear Foreground 7E, 18 / 18 The line above the cursor becomes a scale (1–80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Graphics Menu		Line 23 displays upper and lower case characters and line 24
Foreground Follows 7E, 1F / 1F Background Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1-80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	Line Transmit	7E 14/14	
Background Follows 7E, 19 / 19 Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1-80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01			
Clear Foreground 7E, 18 / 18 Scale / 07 The line above the cursor becomes a scale (1-80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	-		
Scale / 07 The line above the cursor becomes a scale (1-80). This is an aid for graphics and text alignment (local). Roll Up 7E, 01 / 01	-		
for graphics and text alignment (local).	-		The line above the cursor becomes a scale $(1-80)$. This is an aid
Roll Down 7E, 16 / 16	Roll Up		
	Roll Down	7E, 16 / 16	

ADDE	0	1	2	з	4	5	6	7	6	9	Α	В	с	D	Ε	F
0 10 20 30 40 50 50 50 70 40 80 80 80 80 80 80 80	14 0 10 14 0 0 10 10 10 10 10 10 0 0	14 0 10 8 0 10 10 10 10 10 10 0 3	14 0 10 10 10 10 10 10 10 10 10 10 7	14 7C 1C 8 55 7F 7C 1F 7F 7F 1F 7F 1F	14 7C 1C 14 2A 7F 7C 1F 7F 7F 7F 1F C	14 7C 85 7F 7C 1F 7F 7F 7F 1F 18	14 1C 14 0 0 0 0 0 0 1C 1C 1C 1C 1C 38	14 1C 1C 8 0 0 0 0 0 1C 1C 1C 1C 1C 70	14 1C 1C 14 0 0 0 0 0 0 1C 1C 1C 1C 1C 1C 40	14 1C 1C B 0 0 0 0 0 0 1C 1C 1C 1C 1C 1C 40						
0	40	60	70	38	18	C E	E	7	3	1	0	0	0	0	0 E	0
ADD# 	0 0 0 0 0 0 8 8 8 8 0 0 1 4 0			3 1C 0 8 22 1C 0 8 8 8 8 8 8 0 0 4 10	4 1C78 8225778 778 777 777 88			7 3E 8 8 3 0 0 0 0 0 8 8 8 8 8 8 20 2		9 088000000 00008888840 1		B 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0				F 000000000000000000000000000000000000
DF	0	1	2	3	4	5	6	7	8	9	A	B	с 	D	E	F
200 210 220 230 240 250 250 250 270 270 280 280 280 280 280 280 280 280 280 28		00000000000000000000000000000000000000	0 B 14 14 12 14 10 4 2 B 0 0 2 1 1 1 1 1 1 1 1 1 1 1 1 1	0 8 0 3 2 4 4 0 2 2 1 8 0 0 4 -	0 8 0 1 4 1 8 1 0 2 2 2 5 0 8 0 8	0 8 0 3 4 1 2 0 2 8 8 0 0 10	0 0 14 26 20 10 4 0 8 8 0 0 20	0 8 0 0 8 2 A 0 8 8 0 0 8 0 8 2 -	00000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	00000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
ADDR		1	2	3	4	5		7	8	9	A	B	с	D	E	F
300 310 320 330 350 350 350 350 370 380 380 380 380 380 380 380 380 380 38	000000000000000000000000000000000000000	1C B C 3E 4 3E 1C 1C 0 0 4 0 10 1C	2 8 2 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	26 24430 2228 81342 81342	28 C C 2 2 3 4 1 1 0 0 2 0 2 4	38 10 E 2 8 2 0 0 1 3 4 8	2022 22 2022 22 20 20 20 20 20 20 20 20 20 20 20 20 2	1C 3E 1C 4 1C 8 1C 8 1C 0 8 4 0 10 8	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000

ADDR	0	1	2	з	4	5	6	7	8	9	Α	в	с	D	E	F	
400	0	10	22	2E	24	2E	20	1E	0	0	0	0	0	0	0	0	
410	ō	10	22	22	3E	22	22	22	ō	ō	õ	ō	ō	ō	ō	ō	
420	0	30	22	22	30	22	22	ЗC	0	0	0	0	0	0	0	0	
430	0	10	22	20	20	20	22	10	0	0	0	0	0	0	0	0	
440	0	зс	22	22	22	22	22	зс	0	0	0	0	0	0	0	0	
450	0	3E	50	20	30	50	20	ЗE	0	0	0	0	0	0	0	0	
460	0	ЗE	20	20	30	20	20	20	0	0	0	0	0	0	0	0	
470	0	1 C	22	20	20	2E	22	1E	0	0	0	0	0	0	0	0	
480	0	22	22	22	ЗE	22	22	22	0	0	0	0	0	0	0	0	
490	0	10	e	8	8	8	8	10	0	0	0	0	0	0	0	0	
440	0	1E	4	4	4	4	24	16	0	0	0	0	0	0	0	0	
4B0	0	22	24	28	30	28	24	22 3E	0	0	0	0	0	0	0	0	
400 400	0	20 22	20 36	20 2A	20 2A	20 22	20 22	22	õ	õ	õ	ŏ	õ	õ	ŏ	0	
4E0	ŏ	22	22	32	2A	26	22	22	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	õ	
4F0	ŏ	10	22	22	22	22	22	10	ŏ	ŏ	ŏ	ŏ	õ	ŏ	ŏ	õ	
ADDR	0	1	2	3	4	5	6	7	8	9	A	в	с	D	Ε	F	
				~													
500 510	0	3C 1C	22 22	22 22	30 22	20 2A	20 24	20 1 A	0	0	0	0	0	0	00	0	
520	ŏ	30	22	22	30	28	24	22	õ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	õ	
530	ŏ	10	22	20	10	2	22	10	ŏ	õ	ŏ	ŏ	ŏ	õ	õ	õ	
540	ŏ	3E	8	8	ຣັ	8	Ē	ŝ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	
550	õ	22	22	22	22	22	22	10	ō	ō	ō	ō	ō	ō	ō	ō	
560	o	22	22	22	14	14	8	8	0	0	0	0	0	0	0	0	
570	0	22	22	22	2A	2A	2A	14	0	ο	0	0	0	0	0	0	
580	0	22	22	14	8	14	22	22	0	0	0	0	0	0	0	0	
590	0	22	22	22	10	8	8	8	0	0	0	0	0	0	0	0	
5A0	0	3E	2	4	8	10	20 8	3E E	0	0	0	0	0	0	0 0	0	
5B0 5C0	0	20 20	8 20	8 10	8	8 4	2	2	ŏ	ŏ	õ	ŏ	õ	ŏ	õ	0	
500	ŏ	38	8	B	8	8	8	38	ŏ	ŏ	ŏ	ŏ	ŏ	õ	ŏ	ŏ	
5E0	õ	8	10	2A	8	8	8	B	õ	õ	õ	ō	ō	õ	õ	õ	
5F0	õ	õ	8	10	ЗE	10	8	õ	õ	õ	õ	õ	õ	ŏ	õ	õ	
ADDR	0	1	2	3	4	5	6	7	8	9	A	B	c	۵	E	F	
600	10	8	4	0	0	о	o	о	0	0	o	0	0	0	0	0	
610	ō	ō	ō	10	2	1E	22	1E	ō	ō	ō	ō	0	0	o	ō	
620	ō	20	20	20	30	22	22	зс	0	0	o	0	0	0	0	0	
630	ō	o	0	1E	20	20	20	1E	0	0	o	0	0	0	0	0	
640	0	2	2	2	1 E	22	22	1E	0	0	0	0	ο	0	0	0	
650	С	0	0	1 C	22	ЗE	20	1 C	0	0	0	0	0	0	0	0	
660	0	4	8	8	1 C	в	8	8	0	0	o	0	0	0	0	0	
670	0	0	0	1E	22	22	1E	2	10	0	0	0	0	0	0	0	
680	0	20	50	20	зс	22	22	22	0	0	Ő	0	0	õ	0	0	
690	0	8	õ	18	8	8	8	10	0	0	0	0	0	õ	0	0	
6A0 6B0	0	4 10	0 10	4 12	4 14	4 18	24 14	18 12	0	0	0	0	0	0	0 0	0	
600	õ	18	8	6	8	8	8	10	ö	o	õ	ŏ	ŏ	ŏ	ő	õ	
6D0	ŏ	0	õ	36	24	24	2A	2A	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	
6E0	ŏ	õ	õ	30	22	22	22	22	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	
6F0	õ	õ	ō	10	22	22	22	ĩC	õ	ō	ō	ō	ō	ō	ō	ō	
ADDR	0	1	2	3	4	5	6	7	8	9	A	B	с	D	E	F	
700	0	0	0	30	22	22	30	20	20	0	0	0	0	0	0	0	
710 720	0	0	00	1E 16	22 18	22 10	1E 10	2 10	2 0	0	0	ő	00	00	0	00	
730	ŏ	õ	ŏ	18	20	10	2	30	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	
740	ŏ	ě	в	10	8	ê	8	4	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	
750	ŏ	õ	ō	22	22	22	22	ic	õ	ō	ŏ	ō	ō	õ	ŏ	ō	
760	ō	ō	ō	22	22	22	14	8	ō	ō	ò	0	0	0	Ó	ō	
770	0	0	0	55	22	2A	2A	14	0	0	0	0	0	0	0	0	
780	0	0	0	22	14	8	14	22	0	0	0	0	0	0	0	0	
790	0	0	0	22	22	22	1 E	5	10	0	0	0	0	0	0	0	
	ò	0	0	3E	4	6	10	3E	ò	0	õ	0	0	0	0	0	
740	6	8	8	10	20	10	8	8	6	0	0	0	0	0	0	0	
7B0	0 30	8 8	8 8	8 4	0 2	8 4	8 8	8 8	0 30	00	0	00	0	0	0	0	
780 700			ō	4 7F	õ	7F	õ	õ	0	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	
780 700 700		0					-			-						-	
780 700	° °	0 1C	ЗE	36	22	22	36	ЗE	1 C	0	0	o	0	0	0	0	

	TARPLEX	MACRO-ASSEMBL	ER V2.0		PAGE	1	
0.0					27001		
			;May 19	TITLE C	1001		
	0000 '		ASEG	55			
	0000		ORG	00000h			
			consta				
	000A		LF	equ	OAh		
	OOOD		CR	equ	ODh		
	0020		SPC	equ	020h		
	007E		LINC	equ	07Eh	;leadin code	
	0060		RWRG	equ		-low RR1+2	
	0003		KULCDE	e ຊ ບ		skbd unlock code	
			; I/O po				
	0080		KBDPRT	equ	080h	; keyboard	
	0040		ROWPRT	equ		;row interrupt	
	0040		VERPRT	equ		vertical interrupt	
	0040		SETSW	equ		;baud_sel,autolf,u/lcase	
	0002		LCLIND	equ	2	;local indicator	
	0001		BELPRT	equ	1	;bell	
			;ace				
	9000		ACEDTA	equ	07000h	; data	
	9001		ACEITR	equ	ACEDTA+1	;interrupt mask	
	9003		ACECTL	equ	ACEDTA+3		
	9005		ACESTU	equ	ACEDTA+5	;transmit status	
				signment		• · · ·	
	3FFF		FROWH	equ		;first row reg pair	
	3FFE		FROW	equ	FROWh-1		
	3FFD		LROWH	equ		;last row reg pair	
	3FFC		LROW CROWH	equ	FROWh-3		
	3FFB 3FFA			equ	FROWN-4 FROWh-5	cursor row reg pair	
	3FF9		CROW	equ			
	3FF8		CUR	equ	CROW-2	;cursor reg pair	
	3FF7		TOPH	equ		, ;top of page reg pair	
	3FF6		TOP	equ	CUR-2	; cop of page reg pair	
	3FF5		NRW	equ		, ;row counter	
	3FF4		VCALEN	equ		vert calc routine enable	
	3FF3		GSYMBL	equ		;graphics symbol	
	3FF2		AULF	e ຊ ບ		;auto linefeed,O=auto lf	
	3FF1		LOCLM	equ		;local mode,O=remote	
	3FF0		ULCASE			;upper/lower case,O=lower	
	3FEF		GECNTL	equ		;graphic enable,O=disable	
	JFEE		KBDLCK	equ		;keyboard lock,O≕unlock	
	3FED		RTECTL	equ	KBDLCK-1	cursor blink rate cntl	
	3FEC		CUREN	equ	KBDLCK-2	;cursor enable,O=off	
	3FEB		CURTMR	equ		cursor blinking timer	
	3FEA		FFWCT	equ	KBDLCK-4	ace fifo word count	
	3FE9		LEADIN	equ	KBDLCK-5	;leadin mode,O≕no leadin	
	3FE8		ICMD	equ		; insert char mode, O=insert	
	3FE7		CPYCTL	ອຊຸບ		row copy direction cntl	
	3FE6		FFWRT	equ		;fifo write pointer	
	3FE5		FFRD STK	equ		;fifo read pointer ;stack 3FE4h down	
	3FE5 3FA3		LINP	equ		;stack 3FE4h down ;leadin parameter storage	
	3FA2		LINWCT	equ		;leadin word count	
				Equ	IT END 5		
S	TARPLEX	MACRO-ASSEMBL	ER V2.0		PAGE	2	
CR	T801						
	3FA1		LINFH	equ		;leadin func jmp addr high	
	3FA0		LINF	equ	FFEND+1	;leadin func jmp addr low	
	3F9F		FFEND	equ		;ace fifo end	
	3F50		FFSTRT	equ		ace fifo start 3F50/3F9Fh	
	3F20		FBG	equ		;fore/background cntl	
	3F00		DMYROW	equ		idummy row 3F00/3F4Fh	
	3780		FCHR2	equ	03780h	;page 2, 1st char	
				INTO			
) INTERR			***	
	0000	F3	START:		*******	;restart 0	
	0001	21 3F00	DIANT.	LXI	H, DMYROW		
	0004	F9		SPHL		ACT, NON AIGED PAIN	
	0005	C3 00E2		JMP	INIT		
			;				
			, irow int				
	0008	F5	ROW:	PUSH	PSW	;restart 1	
	0009	oc		INR	С		
	000A	C2 0066		JNZ	NOWRAP	;no wrap arroud	
	000D	C3 0061		JMP	VRWRAP	;do wrap arround	
			•				
				al intern			
	0010	E5	VERT	PUSH		;restart 2	
	0011	21 3FF5		LXI	H, NRW		
	0014	4E		MOV	C, M	;load NRW	
	0015	C3 07AE		JMP	VTSUB		
			;				TL/F/5869-7
							12/1/3008-7

0018	C3 0038	ace du ACEDUP:	plicate JMP	ACE	t ;restart 3	
		; (FUNCT	ION} di	sable cur	50r	
001B 001E	21 3FEC 77	DICUR	MOV	H, CUREN	;disable cursor ;a=0	
001F	C9		RET			
		; ; keyboa				
0020	E5	KBD:	PUSH	н	;restart 4	
0021 0022	F5 21 3FEE			PSW H, KBDLCI	K : kouboond look oot	1
0025	C3 02C9		JMP	KBDINT		1
				interrup		
0028	F5	ROWDP:	PUSH	PSW	;restart 5	
0029	0C C2 0066		INR JNZ	C NOWRAP		
002A 002D	C3 0061		JMP	VRWRAP		
				icate int		
0030	C3 0010			VERT		
0033	EB	TABSTP:	XCHG		; hl=crow	
0034 0035	D1 D1		POP POP	D D	;remove call	
0035	73		MOV	M, E	;return org crow	
0037	C9		RET		-	
		; PAGE				
STARPLE	X MACRO-ASSEN			PAGE	3	
CRT801		DELIC VE. U		1 HOL	5	
0038		ACE IN	TERRUPT			
		; ******	*****	******	******	
0038	E5	ACE:		H		
0039 003A	F5 21 3FEB		PUSH LXI	PSW H, CURTMI		
0030	D5		PUSH	D	`	
003E	AF		XRA	A		
003F	77		MOV	M, A	reset cursor timer	
0040 0041	2B 34		DCX INR	H M	;FFWCT ;fifo_empty?	
0042			JNZ	STFIFD	inc, store to fifo	
0045	22		DOV			
0045 0046	2B B6		DCX	H M	;leadin	
0047	3A 9000		LDA		read ace	
004A	FB		EI			
004B	C2 0283		JNZ	LINMDE.	;leadin mode	
004E	E6 7F	FIFACE:		07Fh		
0050 0053	11 0267 D5		LXI PUSH	D, FFCHK	;pseudo call	
0054	FE OA		CPI	LF	, pseudo cari	
0056	CA 0381		JZ	LFEED		
0059	CD 015C		CALL	CALJMP		
0050	E6 87 F8		ANI		;leadin+ymp addr high	
005E 005F	67	LCLFUN	RM MOV	H. A	;leadin required, return ;jmp addr high	
0060	E9	2021 011	PCHL		do function	
				continue		
0061	3E 01	,****** VRWRAP	******* MVI	A, 1	***	
0063	32 5000	VRWRAP:	STA	05000h	;wrap arround addr	
0066	D3 40	NOWRAP:	OUT	ROWPRT	;clr row flip/flop	
0068 0069	F1 FB		POP EI	PSW		
006A	C9		RET			
		WRITE				
006B	GE OD	; ****** SNDCR:		********* A, CR	********	
006D	57	WTACEA:		D, A		
006E	3A 9005	WTACED:		ACEST	;check status	
0071	FE 60	HINCED:	CPI	060h	;hold/tx register	
0073	DA 006E		JC	WTACED	inot ready	
0076	CD 075E		CALL	\$DLY	;delay	
0079	7 A	DUTACE:		A, D		
007A	32 9000 C9		STA RET	ACEDTA	;write to ace	
0070			061			
007D	67	PAGE				т

CRT801	MACRO-ASSEMBLI	ER V2.0		PAGE 4	
007E		ROW ST	ART LOOK	UP TABLE (start addr≖7Eh)	

007E	3EBO	ROW47D:		O3EBOh	
0080 0082	3000 3050	ROW48D: ROW1:	dω	03000h 03050h	
0082	3030 30A0	ROW2:	dw dw	030A0h	
0086	30F0	ROW3:	dω	030F0h	
0088	3140	ROW4:	đw	03140h	
008A	3190	ROW5:	dω	03190h	
0080	31E0	ROW6:	d w	031E0h	
008E 0090	3230 3280	ROW7: ROW8:	dw dw	03230h 03280h	
0070	3200	ROW9:	dω	032D0h	
0094	3320	ROW10:	dw	03320h	
0096	3370	ROW11:	dω	03370h	
0098	3300	ROW12:	dω	033C0h	
009A	3410	ROW13:	dw.	03410h	
009C 009E	3460 34b0	ROW14: ROW15:	d ພ d ພ	03460h 03480h	
0040	3500	ROW15:	dω	03500h	
0042	3550	ROW17:	dω	03550h	
00A4	35A0	ROW18:	dω	035A0h	
0046	35F0	ROW19:	dw	035F0h	
BAOO	3640	ROW20:	dω	03640h	
00AA	3690	ROW21:	dω	03690h	
00AC 00AE	36E0 3730	ROW22: ROW23:	dw dw	036E0h 03730h	
0080	3780	ROW23	dw	03780h	
0082	3700	ROW25:	dω	037D0h	
0084	3820	ROW26:	dω	03820h	
0086	3870	ROW27:	dω	03870h	
OOBB	3800	ROW28:	đω	038C0h	
OOBA	3910	ROW29:	d w	03910h	
OOBC OOBE	3960 3980	ROW30: ROW31:	d ພ d ພ	03960h 03980h	
0000	3A00	ROW32:	dω	OBAOOh	
0002	3A50	RDW33:	dω	03A50h	
0004	3AA0	ROW34:	dω	OBAAOh	
0006	3AF0	ROW35:	dω	O3AFOh	
0008	3840	ROW36:	dω	03B40h	
000A	3B90	ROW37: ROW38:	d ພ d ພ	03890h 038E0h	
00CC 00CE	3BE0 3C30	ROW38:	d យ	03C30h	
OODO	3080	ROW40:	dw	OGCBOh	
0002	3CD0	ROW41:	dω	O3CDOh	
OOD4	3D20	ROW42:	dω	03D20h	
00D6	3D70	ROW43:	dω	03D70h	
OODB	3DCO	ROW44:	đu	OBDCOh	
OODA OODC	3E10 3E60	ROW45: ROW46:	ದ ಬ ದ ಬ	03E10h 03E60h	
QODE	3EB0	ROW47:	ປພ	OSEBON	
00E0	3F00	ROW48:	dω	03F00h	
		PAGE			
STARPLEX CRT801 00E2	MACRO-ASSEMBL	ER V2.0		PAGE 5	
		; INITIA			
00E2	3E 20			**************************************	
00E2 00E4	3E 20 16 E9	INIT:	MVI MVI	D, low LEADIN; byte count	
OOE6	CD 04C7		CALL	DRLLP ;store spaces	
00E9	AF		XRA	A	
OOEA	32 3FA2		STA	LINWCT ;zero leadin word count	
OOED	16 17		MVI	D,256-low LEADIN; byte count	
OOEF	CD 04C7		CALL	DRLLP ; store zeros	
00F2 00F5	31 3FE7 21 5050			SP, STK+2 H, 05050h	
OOFB	E5		PUSH	H ;set up fifo rd/wrt ptrs	
00F9	CD 04AE		CALL	CURULK ; enable cursor, unlock kbd	
OOFC	32 3FEA		STA	FFWCT ; zero fifo word count(FFh)	
OOFF	23		INX	H ; RTECTL	
0100	36 1C		MVI	M,O1Ch ; cursor blink cntl	
0102	2E FC		MVI	L,low LRDW;last row M,low RR24	
0104 0106	36 BO 2E FE		MVI MVI	m, low RR24 L, low FROW; first row	
0108	36 82		MVI	M, low RR1	
0100	DB 80		IN	KBDPRT ;clear keyboard intr	
0100	CD 04CE		CALL	CLRSCN ; clear screen	
010F	CD 07BF		CALL	ACESW ; init ace, read setsw	
0112	3E 3F	PATTN:	MVI	A, O3Fh	
0114 0117	21 3780 75	PTNLP:	LXI MOV	H,FCHR2 ;1st byte of page 2 M,L ;write pattern	
0118	23		INX	H	
0119	BC		CMP	н	
				Т	FL/F/5869-9

011A	C2 0117		JNZ	PTNLP		
011D	D3 01		OUT		;ring bell for ready	
0.115					AFTER VERTICAL INTERRUPT	
011F 0121	2E EB 7E	VCAL	MVI MOV	L, low CU A, M	URINK	
0122	2F		CMA			
0123	34		INR	м	;cursor timer	
0124	23		INX	н	;cursor enable	
0125	A6			M		
0126 0127	23 A6		INX ANA	H M	;rate cntl ;blink rate mask	
0128	C4 0174		CNZ		CUTSOT ON	
012B	3E 20		MVI		;8350 offset	
012D	84		ADD	н	;offset addr high	
012E	67		MOV	H, A	;=5Fh if cursor off	
012F 0130	E 5 2A 3FFE		PUSH LHLD	H FROW	;save cursor	
0133	2B		DCX	н	;fetch row start	
0134	7E		MOV	A, M		
0135	C6 20		ADI	020h	;offset addr high	
0137	57 2B		MOV	D, A		
0138	28	PAGE	DCX	н		
	MACRO-ASSEMBLI	ER V2.0		PAGE	6	
CRT801 0139						
0139	5E		MOV	E, M	;de=top of page	
013A	7D		MOV	A, L	;calc row wrap constant	
013B	OF		RRC		; divide by 2	
0130	C6 BF		ADI	OBFh	; add offset	
013E	E1		POP DI	н	cursor location	
013F 0140	F3 22 3FF8		SHLD	CUR	jupdate cursor	
0143	EB		XCHG			
0144	22 3FF6		SHLD	TOP	;update top of page	
0147	21 3FF5		LXI	H, NRW		
014A	77		MOV	м, А Н	;row wrap constant	
014B 014C	2B 36 00		DCX MVI	M. 0	;VCALEN ;disable VCAL routine	
014E	FB	WAIT:	EI		, disobie vone roovine	
014F	76		HLT			
0150	7E		MOV	A, M	VCALEN	
0151	B7		ORA	A	check from vert intr	
0152	CA 014E C3 011F		JZ JMP	WAIT VCAL	;no ;do screen calculations	
0100	00 011		011	10112		
			ATE JMP	ADDR		
0158	2E E9	CJMP:	MVI	L, low LE		
015A 015C	E6 7F FE 20	CALJMP:	ANI	07Fh SPC	;mask 1st bit	
0155	DA 0168	CHEORY.	JC		;O-1Fh, func	
0161	FE 7E		CPI	07Eh		
0163	DA 0372		JC	CHAR	;20-7Dh, char input	
0166	D6 5E	-	SUI	05Eh	7E/7Fh to 20/21h	
0168 0169	2B 74	FUNC:	DCX MOV	н м,н	;insert mode ;h<>0,defeat insert mode	
016A	07	JMP ADD:			;#2, msb=0	
016B	5F		MOV	E, A	;d=02h (jmp tbl)	
0160	1A		LDAX	D	;fetch jmp addr low	
016D 016E	6F 13		MOV INX	L, A D		
016F	13 1A		LDAX	a	;fetch jmp addr high	
0170	C9		RET	-	,	
0.17/	05 50				CUR TO END DIFF	
0171 0173	3E 50 90	DFCLOC	SUB	A, 80 B	cursor to end difference	
01/0	,0		000	5	Jeonson vo end airrevence	
				DR LOCATI	ION	
0174	2A 3FFA	CURLOC		CROW		
0177	20		DCX MOV	H		
0178 0179	56 2B		DCX	D'W D'M		
017A	5E		MOV	E, M		
017B	68		MOV	L, B		
017C	19		DAD	D	;hl≃cursor address	
017D	C9	PAGE	RET			
		FAGE				
	MACRO-ASSEMBLE	ER V2.0		PAGE	7	
CRTB01						
017E		։թղա սթ			TABLE (start addr=17Eh)	
					IABLE. (Start ador=1/En) ####################################	
017E	00E0	RR47D:	đw	OOEOh	· · · · · · · · · · · · · · · · · · ·	
0180	0082	RR48D	dω	0082h		
0182	0084	RR1:	dω	0084h		TL/F/5869-10
						12/7/0009-10

2 • • 2	000	a	20271	
0184 0086	RR2:	d w atus	0086h	
0186 0088	RR3:	d ឃ 	0088h	
0188 008A	RR4:	dω	OOBAh	
018A 008C	RR5:	dw d.	008Ch	
01BC 008E	RR6:	dw d	008Eh 0090h	
018E 0090	RR7:	dω		
0190 0092	RRB:	dω	0092h	
0192 0094	RR9:	dω	0094h	
0194 0096	RR10:	dω	0096h	
0196 0098	RR11:	dω	0098h	
019B 009A	RR12:	dω	009Ah	
019A 009C	RR13:	dω	009Ch	
019C 009E	RR14:	dω	009Eh	
019E 00A0	RR15:	dω	00A0h	
01A0 00A2	RR16:	dω	00A2h	
01A2 00A4	RR17:	dω	00A4h	
01A4 00A6	RR18:	dω	00A6h	
01A6 00A8	RR19:	dω	OOABh	
01AB 00AA	RR20:	dω	OOAAh	
01AA OOAC	RR21:	dω	OOACh	
01AC OOAE	RR22:	dω	OOAEh	
OIAE OOBO	RR23:	dω	OOBOh	
01B0 00B2	RR24:	dω	00B2h	
01B2 00B4	RR25:	dω	00B4h	
01B4 00B6	RR26:	dω	OOB6h	
01B6 00B8	RR27:	dω	OOBBh	
01BB OOBA	RR28:	៨ឃ	OOBAh	
O1BA OOBC	RR29:	៤ អ	OOBCh	
O1BC OOBE	RR30:	dω	OOBEh	
OIBE OOCO	RR31:	dω	OOCOh	
01C0 00C2	RR32:	dω	00C2h	
01C2 00C4	RR33:	dω	00C4h	
01C4 00C6	RR34:	dω	00C6h	
0106 0008	RR35:	dω	OOCBh	
01C8 00CA	RR36:	dω	OOCAh	
OICA OOCC	RR37:	dω	OOCCh	
01CC OOCE	RR38:	dω	OOCEh	
OICE OODO	RR39:	dω	OODOh	
01D0 00D2	RR40:	dω	00D2h	
01D2 00D4	RR41:	dω	OOD4h	
01D4 00D6	RR42:	dω	00D6h	
01D6 00D8	RR43:	dω	OODBh	
01DB 00DA	RR44:	dω	OODAh	
	RR45:	ປພ	OODCh	
01DA OODC				
01DA OODC 01DC OODE	RR46	dω	OODEh	
			00DEh 00E0h	
01DC OODE	RR46:	dω		
01DC OODE 01DE OOEO	RR46: RR47:	dw dw	OOEOh	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084	RR46: RR47: RR48: RR1D:	dw dw dw	00E0h 0082h 0084h	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI	RR46: RR47: RR48: RR1D:	dw dw dw	00E0h 0082h	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084	RR46: RR47: RR48: RR1D: EMBLER V2.0	dw dw dw dw	00E0h 0082h 0084h PAGE B	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI	RR46: RR47: RR48: RR1D: EMBLER V2. O	dw dw dw dw DRESS	OOEOh OOB2h OOB4h PAGE B CONSTANTS	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI	RR46: RR47 RR48: RR1D: EMBLER V2: 0 ; JUMP AI ; ******	dw dw dw dw DDRESS	00EOh 0082h 0084h PAGE B CONSTANTS	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; JUMP AI ; J15	dw dw dw dw DDRESS ****** : - 0=n	00EOh 0082h 0084h PAGE B CONSTANTS ******	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12	dw dw dw DDRESS ****** :- O=n 2: - par	OOEOh OOB2h OOB2h PAGE B CONSTANTS ************************************	
01DC OODE 01DE OOEO 01EO 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ****** ; A15 ; A14/A12 ; A11	dw dw dw dw DDRESS ******* :- 0=n 2: - par :- 0=1	OCEON OOB2h OOB4h PAGE B CONSTANTS ************************************	
01DC OODE 01DE OOEO 01EO 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; 415 ; A15 ; A14/A12 ; A11 W1	dw dw dw dw DDRESS :- 0=1 :- 0=1 equ	OOEOh OOB2h OOB2h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; #****** ; A15 ; A14/A12 ; A11 W1 W2	dw dw dw dw DDRESS ******* :- O=n :- O=1 equ equ	OGEOh OOB2h OOB2h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 B000	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AL ; #****** ; A15 ; A14/A12 ; A11 W1 W2 LIN	dw dw dw DRESS ******* :- 0=n 2:- par :- 0=1 equ equ	OCEON OOB2h OOB4h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; #****** ; A15 ; A14/A12 ; A11 W1 W2	dw dw dw dw DDRESS ******* :- O=n :- O=1 equ equ	OCEOh OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 B000	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AL ; #****** ; A15 ; A14/A12 ; A11 W1 W2 LIN	dw dw dw DRESS ******* :- 0=n 2:- par :- 0=1 equ equ	OCEON OOB2h OOB4h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; #****** ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC	dw dw dw DDRESS ******* :- O=n 2: - D=1 equ equ equ equ	OCEON OOB2h OOB4h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 085F	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; #****** ; A15 ; A14/A12 ; A11 W1 W1 W2 LIN NLC F1	dw dw dw dw DDRESS :- 0=n 2: - par :- 0=1 equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800 0A5F 03B9 03B1	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W1 W2 L1N NLC F1 F2	dw dw dw DDRESS :- 0=n 2: - 0=1 equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 B000 0800 0800 085F 03B9 03B1 0C34	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; #****** ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3	dw dw dw dw DDRESS ******* :- 0=n 2: - par :- 0=1 equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; #****** ; A15 ; A14/A12 ; A11 W2 LIN NLC F1 F2 F3 F3 F4 F5	dw dw dw dw DDRESS ******* :- O=n 2:- par :- O=1 :- O=1 equ equ equ equ equ equ equ equ equ	OGEOh OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F3 F4 F5 F6	dw dw dw dw DDRESS ******* :- 0=n 2:- par :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; #****** ; A15 ; A14/A12 ; A11 W2 LIN NLC F1 F2 F3 F3 F4 F5	dw dw dw dw DDRESS ******* :- O=n 2:- par :- O=1 :- O=1 equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F4 F5 F6 F7	dw dw dw dw DDRESS ******* :- 0=n 2:- par :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0ASF 03B9 03B1 0C34 060A 0437 041F	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F4 F5 F6 F7	dw dw dw dw DDRESS :- 0=n :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800 0800 085F 0389 0381 0C34 060A 0437 041F 844E	RR44: RR47: RR48: RR1D: =MBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10	dw dw dw dw DDRESSS ******* :- 0=n 2: - 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0A5F 03B1 0C34 060A 0437 041F 844E 8440	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F4 F5 F6 F7	dw dw dw dw DRESS :- 0=n 2: - 0=n 2: - 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 085F 03B1 0C34 060A 0437 041F 844E 8460 8472 8472	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; *******; ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F4 F5 F6 F7 F8 F9 F10 F11 F12	dw dw dw dw DDRESS ******* - O=n 2: - par :- O=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11	dw dw dw dw DDRESS :- 0=n 2: - 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0092 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0850 0A5F 03B9 03B1 0C34 060A 0437 041F 844E 8460 84D8 85F3 8472 848C	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13	dw dw dw dw DRESS :- 0=n 2: - 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOBAH PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: SMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14	dw dw dw dw DRESS ***-0=n 2:- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 B000 0800 0A5F 03B9 03B1 0C34 060A 0437 041F 844E 8460 84D8 85F3 8472 848C 84AE 80D1	RR44: RR47: RR48: RR1D: =MBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15	dw dw dw dw DRESS :- 0=n 2: - 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOBAN PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: SMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14	dw dw dw dw DRESS ***-0=n 2:- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F14 F15 F16	dw dw dw dw dw concernent dw dw dw concernent concernen	OCEON OOB2h OOB2h OOB4h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800 045F 03B9 03B1 0C34 060A 0437 03B1 0C34 060A 0437 041F 844E 8460 845F3 8472 848C 84AE 84AE 84AE 84AE	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ****** ; A15; ; A14/A12 ; A11 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17	dw dw dw dw DRESS :- 0=n :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F14 F15 F16	dw dw dw dw dw concernent dw dw dw concernent concernen	OCEON OOB2h OOB2h OOB4h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: =MBLER V2.0 ; JUMP AI ; ****** ; A15 ; A14/A12 ; A11 W2 LIN W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18	dw dw dw dw dw DRESS :- 0=n 2: - 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOBAH PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0092 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 045F 03B9 03B9 03B9 03B9 03B9 03B9 03B9 03B9	RR46: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15; ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18 F19	dw dw dw dw dw DRESS :- 0=n :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOBAH PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0600 0437 0437 0431 0C34 060A 0437 041F 844E 8446 8472 8472 8472 8472 8472 8472 8472 8472	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18 F17 F19 F20	dw dw dw dw dw concernent dw dw dw concernent concernen	OCEON OOB2h OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRD-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 008000000	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15; ; A14/A12 ; A11 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18 F19 F20 F21	dw dw dw dw dw DRESS :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOBAN PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 045F 03B1 0C34 060A 0437 041F 844E 8460 84D8 85F3 8472 848C 84AE 84AE 84AE 84AE 84AE 84AE 84AE 84AE	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18 F19 F20 F21 F22	dw dw dw dw dw DRESS ***- 0=n 2:- par :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB2h OOB3h PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 0800 0800 0800 08	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ****** ; A15 ; A14/A12 ; A11 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18 F19 F20 F21 F22 F23	dw dw dw dw dw DREESS :- 0=n equ equ equ equ equ equ equ equ equ equ	OCEON OOBAN PAGE B CONSTANTS ************************************	
01DC 00DE 01DE 00E0 01E0 0082 01E2 0084 STARPLEX MACRO-ASSI CRTB01 1000 2000 8000 0800 045F 03B1 0C34 060A 0437 041F 844E 8460 84D8 85F3 8472 848C 84AE 84AE 84AE 84AE 84AE 84AE 84AE 84AE	RR44: RR47: RR48: RR1D: EMBLER V2.0 ; JUMP AI ; ******* ; A15 ; A14/A12 ; A11 W1 W2 LIN NLC F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16 F17 F18 F19 F20 F21 F22	dw dw dw dw dw DRESS ***- 0=n 2:- par :- 0=1 equ equ equ equ equ equ equ equ equ equ	OCEON OOB2h OOB2h OOB3h PAGE B CONSTANTS ************************************	TL/F/5869-11

0419	F23	5 в	eq u	DEL		
83DB	F24	6 в	qu	DCROW+LIM	N	
8400	F23	7ε	equ .	DRTLN+LIM	N	
B4BA	F28	B e	ะอุบ	DRTPG+LI	N	
8642	F29	96		CFB+LIN		
84CE	F30	0 6	equ	CLRSCN+LI	IN	
8086	F3:			KBLK+LIN+		
BOIB	F3			DICUR+LIM		
871D	F3:			SNDLNE+L		
878F	F34			ACESW+LI		
OAC5	F3:			LINSET+NL		
B6BC	F3		equ	DEGRPH+L1		
	PAG	GE .				
STARPLEX	ACRO-ASSEMBLER	V2.0		PAGE 4	9	
CRT801						
01E4						
	; PI	UT A WO	DRD TO A	CE		
01E4 5		DACE		D, A		
	3A 9005		_DA	ACESTU		
	E6 20		ANI	020h		
	CA 01E5	,	JZ	KBDACE+1;	not ready	
01ED C	3 0079	,	JMP	OUTACE		
				ONSTANTS		
01F0 0			iω	1707 ;		
	D139 B6		tω	313 ;		
			ωt		0.3% 2	
			υt		0.3% 3	
01F8 0			υi		0.3% 4	
		600: i	វយ		2.3% 5	
		9200: 0			; 2. 3% 6	
01FE 0	0005 B3	8400: 0	lω	5 :	2.3% 7	
			N JUMP T		(start addr=200h)	

		LJMP: (null/toggle local	
	3472				A:roll up	
	FEBA CB		iω		Bigraphics mode	
	B4AE		lω ·		C:on cursor/unlock kbd	
	3358		1w		;D:strip off character	
	BDD1 CE		iω tu		E:read cursor	
	301B		1w 1		; F: disable cursor	
	DC34 CG D41F		វយ វយ		;G:bell/scale ;H:cursor left	
	060A		ίω		;I:tab	
	03B1		lω		Jline feed	
	3460		ว่า ม่าม		K: cursor down	
	3460 344E		tw		L: cursor up	
	0389		lω		M:carriage return	
	35F3		tω		N:next page	
	B4C0		lω		; O: delete rest of line	
	0437		dω		;P:cursor right	
0222 4	ADA2 CQ	: 0	dω	F16	;Q:address cursor/menu	
	B4D8		tω	F10	R:home cursor	
0226 8	3512		dω	F22	S:strip off a line	
	371D		វ័យ		;T:transmit a line	
	BCB6 CU		dω		;U:lock keyboard	
0220 8	B48C		dω		;V:roll down	
	B4BA		ជ័យ		;W:delete rest of page	
	B642		dω		;X:clear fore/background	
	B4F7		dω		;Y:background follows	
	8505		dω		Zinsert line	
	B6BC		៨ ឃ		;1B:esc/defeat graphics	
	B4CE		៨ឃ		;1C:home and clear screen	
	AD26 C1		dw 		;1D:insert/strip lne/rng	
	B36B		dw d	F19	;1E: insert character	
	B4F1		dw dw	F17 F35	;1F:foreground_follows ;7E:leadin/ace,u/l-case	
	0AC5 C7		dw dw		; 7F: delete	
0242 (0419		dω	F25	, ,, , deleve	
STARPLEX	MACRO-ASSEMBLER	V2. 0		PAGE	10	
CRT801						
	; 5	TORE A	WORD TO	ACE FIF	0	
0244		FIFO:		A, 80		
	BE		CMP	Μ	;exceeding 80 words?	
	3A 9000	1	LDA	ACEDTA	iread ace	
	DA 032C		JC	OVRNG	;more than 80 words	
	2E E6		MVI	L, low FF	WRT	
	54		MOV		;set up write pointer high	
	5E		MOV		;set up write pointer low	
	12		STAX		;store to fifo	
	7B		MOV	A, E		
	30		INR		advance pointer	
	FE AO		CPI		D+1;exceeding BO words? ;less than BO words	
	DA 025A		JC RAR			
0259	1F	1	R 11 R		;fifo start again	TL/F/5869-12
						12/1/3003-12

025A 025B	77 FB	WFFRNG:	MOV EI	M, A	;advance write pointer	
0250	D1	CMRTN:	POP	D		
025D	F1		POP POP	PSW		
025E	E1		FUF	н		
			ION} unu	sed keys		
025F	C9	RTN:	RET			
				RN, addr		
0260 0261	AF 32 3FEE	KLCRTN:	XRA STA	A KBDLCK	;enable keyboard	
0261	C3 025C		JMP	CMRTN		
00/7	D1 0554				addr high≕⊋h ;fifo word count	
0267 026A	21 3FEA 35	FFCHK:	LXI DCR	M	;fifo empty?	
026B	FA 025C		JM	CMRTN	;empty	
		READ A	WORD FR	DM ACE F	IFO	
026E	2E E5	RDFIFO:	MVI	L, low Fi		
0270	54 5E		MOV MOV	D, H	;set up read pointer high	
0271 0272	7B		MOV	E, M A, E	;set up read pointer low	
0273	30		INR	A	;advance read pointer	
0274	FE AO		CPI		ND+1; exceeding 80 words?	
0276 0279	DA 027A 1F		JC RAR	RFFRNG	;less than 80 words ;fifo start again	
027A	77	RFFRNG		M, A	store read pointer	
027B	2E E9		MVI	L, low L	EADIN	
027D 027E	7E 87		MOV ORA	A, M A	ileadin mode≖0?	
027E	1A		LDAX	D	read fifo word	
0280	CA 004E		JZ	FIFACE	;not leadin, normal entry	
		PAGE				
	MACRO-ASSEMB	LER V2. 0		PAGE	11	
CRT801						
0283		LEADIN	MODE			
0283	E6 7F	LINMDE		07Fh	;mask input	
0285	11 0267		LXI	D, FFCHK		
0288	D5		PUSH	D	;pseudo call	
0289 0288	2E A2 5E		MVI MOV	L, low L E, M	;leadin word count	
0280	1D		DCR	ε	;word count=0?	
028D	F2 02B3		JP	LINPRA	parameter entries	
0290 0292	FE 20 D2 02C2		CPI JNC	SPC ILELIN	;control codes? ;not cntl code; error	
0295	CD 016A		CALL	JMPADD	fetch jmp address	
0298	B7		DRA	Α		
0299 0290	F2 02C2 E6 77		JP ANI	ILELIN 077h	;code requires no leadin ;word count/jmp addr high	
029E	67		MOV	H, A	;save	
029F	E6 70		ANI	070h	imask word count	
02A1	CA 02BE		JZ	LINEXE	;do function	
02A4	OF	LINPFN:	RRC		right justify word count;	
02A5	OF		RRC			
02A6 02A7	OF OF		RRC RRC			
0247	32 3FA2		STA	LINWCT	;store to word count reg	
02AB	3E 07		MVI	A, 07h	,	
02AD	A4		ANA	н	;get jmp addr high	
02AE 02AF	67 22 3FA0		MOV SHLD	H, A LINF	;save function jmp addr	
0282	C9		RET		, , Gilberton, Jimp Gool	
	70		MOU	M, E	1	
02B3 02B4	73 16 00	LINPRA:	MVI	D, O	;leadin word count−1 ;d=0,e=word count−1	
0286	23		INX	Ĥ	; LINP	
0287	19		DAD	D	;hl=para pointer	
02B8 02B9	77 CO		MOV RNZ	M, A	;store word ;word count<>0, next word	
02BA	2A JFAO		LHLD	LINF	;load leadin jmp address	
02BD	AF		XRA	Α		
02BE	32 3FE9	LINEXE	STA		;defeat leadin	
0201	E9	LINEXE.	PCHL	LENDIN	ido function	
		ILELIN:				
			UNT	BELPRT	;illegal code after leadin	
02C2 02C4	D3 01 AF	1000.414.	XRA		;a=0, reset leadin	

0205	32 3FE9	LINSET:		leadin a LEADIN	node ;a<>0, set leadin	
0208	C9		RET			
		PAGE				
STARPLEX	MACRO-ASSEMBLI	ER V2.0		PAGE	12	
CRTB01						
0209		; keyboar	rd intern	rupt cont	inue	
0209	AF	KBDINT:		A		
02CA 02CD	32 3FEB 86		STA ORA	CURTMR M	<pre>;reset cursor blink timer ;keyboard locked?</pre>	
02CE	DB 80		IN		;read keyboard	
02D0 02D1	FB D5		E I PUSH	D		
02D2	C2 0327		JNZ	-	;keyboard not active	
0205	87		DRA	A		
0206	CA 0333		JZ	TGLCL	;toggle local	
0.000	77		MOV	M A	;defeat graphics	
02D9 02DA	77 11 0260			M, A D, KLCRTM	;lock keyboard N	
02DD	D5		PUSH	D	;generate pseudo call	
02DE	23		INX	н	GECNTL	
02DF	B6		ORA	м		
02E0 02E1	23 FE 61		INX CPI	H 061h	; ULCASE	
02E1	DA O2EC		JC	NLCSE	;not lower case	
02E6	FE 7B		CPI	07Bh		
02E8 02EB	D2 02EC 96		JNC SUB	NLCSE M	;not lower case ;u/l case cntl,m=20h/0	
		NI 005				
02EC 02ED	23 5E	NLCSE	INX MOV	H E,M	;local	
02EE	10		INR	E	;local mode?	
02EF	C2 01E4		JNZ	KBDACE	;write to ace	
02F2	FE AO	LCL:	CPI	OAOh	;parameter entry?	
02F4	D2 068D		JNC	LGPARA	; yes	
02F7	CD 0158		CALL	CUMP	;get jmp addr	
02FA	E6 OF		ANI	OFh	1110	
02FC 02FE	FE 08 DA 005F		CPI JC	08h LCLFUN	;local? ;do local function	
0301	7B		MOV	A, E	;read lookup tbl ptr	
0302 0304	FE 05 CA 06B7		CPI JZ		;entl B? ;enable graphics mode	
0307	FE OB		CPI	low CE+1	l)cntl E?	
0307 0300	CA 0773 FE OF		JZ CPI		;display cursor location l;cntl G?	
030C	CA 0784		JZ	SCALE	;put scale	
0311	FE 23		CPI		lichtl Q?	
0313 0316	CA 06D5 FE 41		JZ CPI	PGM low C7E⊣	;put graphics menu F1	
0318	CA O7CD		JZ	ATGUL	;init ace,toggle u/l case	
031B 031D	FE 3B CA 034F		CPI JZ	low C1D- ROM2	+1 ;do rom2 functions	
0320	FE 2B		CPI		; cntl U?	
		PAGE				
	MACRO-ASSEMBL	ER V2.0		PAGE	13	
CRT801 0322						
0322	со		RNZ		; unused keys	
0323	D1		POP	D	;pseudo rtn+lock kbd	
0324	C3 0261		JMP	KLCRTN+1		
0327 0329	EE 03 CA 025A	KNACTV:	XRI JZ		;lock/unlock_kbd? ;unlock_keyboard	
0350	3E 50	OVRNG:	MVI	A, 80	; for FFWCT	
032E 0330	D3 01 C3 025A		DUT JMP	BELPRT CMRTN-2	;lock kbd	
0333 0334	23 77	TGLCL	INX MOV	Н М,А	;GECNTL ;disable graphics mode	
0334 0335	77 32 3FE9		STA		joisable graphics mode jreset leadin	
0338	2E F1		MVI	L, low LC		
033A 033B	7E 2F		MOV CMA	A, M	;toggle local	
0330	77		MOV	M, A		
033D 0340	CD 07E8 C2 0345		CALL JNZ	EDACE ONLINE	;en/disable ace	
0343	3E 1E		MVI	A,03h X0	JR O1Dh	
0345	EE 1D	ONLINE:	XRI	O1Dh		TI /F /5000 11
						TL/F/5869-14

0347	00			nop		;out lclprt	
0348	00			nop	DICCT	· · · · · · · · · · · · · · · · · · ·	
0349 034C		3FED 025C		STA JMP	CMRTN	;select blink rate	
0040	00	0200		011	U. I.		
			; JMP TO	ROM2			
034F			ROM2:	LDA		; check presence of rom2	
0352	FE CO	55		CPI RNZ		;=55h? ;not exist	
0354 0355		0801		JMP		jnot exist jok, do jmp	
0000	••	0001		••••		, ok, 10 jp	
0358	cn	0171	STOFCH:			∣ character ;get cur loc and diff	
035B	ЗD	01/1	STUPUR.	DCR	A	,get cor loc and diff	
0350		0434		JZ		;last column, error	
035F	23		STOFLP:		н		
0360	56			MOV	D, M		
0361 0362	2B 72			DCX MOV	н м, D	ido copy	
0363	23			INX	н	/ 48 2009	
0364	ЗD			DCR	A		
0365		035F		JNZ	STOFLP		
0368	сз	03A4		JMP	PSPC	;put a space	
			: {EUNCT	ION} ins	ert chara	acter	
036B	AF		INSCHAR		A		
0360	32	3FE8		STA	ICMD	;enable insert mode	
036F	СЗ	038D		JMP	INSCHR		
			PAGE				
STARPLEX	MAC	RO-ASSEMBLE	ER V2.0		PAGE	14	
 CRTB01							
0372							
0070				TER INPU			
0372 0373	D1 2E		CHAR:	POP MVI	D L, low FE	;pseudo return 10	
0375	ĀE	20		XRA		;add attribute	
0376	AD			XRA		;⊤emove space code	
	-						
0377	2A 2B	3FFA		LHLD DCX	CROW H	;calculate cursor loc	
037A 037B	2B 56			MOV	D, M		
0370	2B			DCX	н		
037D	5E			MOV	E, M		
037E	68			MOV		;row start+cursor	
037F 0380	19 77			DAD MOV		;hl=cursor address ;write to screen	
0380	//			100	107 E	Julite to screen	
0381	ЗE	4F		MVI	A, 79		
0383	AB			XRA		;last column?	
0384		0300	MIDOUD	JZ		ilast, do scroll	
0387 0388	04 34	3FE8	MIDCHR:	LDA		<pre>;else advance cursor ;insert mode?</pre>	
0388	B7	51 20		ORA	A	, inserv mode.	
0380	со			RNZ		;not insert mode	
038D 038F	3E 90	41	INSCHR	SUB	A, 79 B	; byte counter	
038F		OBAB		JZ		; byte counter ; cursor at last column	
0393		3FFA		LHLD	CROW		
0396	5E			MOV	E' M		
0397 0398	23 56			INX MOV	н р, м	;d=row end+1	
0378	1B			DCX		row end	
039A	EB			XCHG			
039B	2B		INSLP:	DCX	H		
039C 039D	56 23			MOV INX	D, M H		
039E	72			MOV		; do copy	
039F	2B			DCX	н	· · · · · · · · · · · · · · · · · · ·	
0340	ЗD			DCR	Α		
03A1	C2	039B		JNZ	INSLP		
03A4	ЗE	80	PSPC:	MVI	A, 080h		
0346	A2			ANA		;get character attribute	
03A7	F6	20		ORI	SPC	;add space	
0349	77			MOV	M, A		
AAEO	C9			RET			
OGAB	зс		ILCHAR:	INR	A	;make a⇔0	
OGAC	DЗ			OUT	BELPRT		
03AE		OBCE		JMP		;defeat insert char mode	
			PAGE				TL/F/5869-15
							,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,

CRTB01	MACRO-ASSEMBL	ER V2.0		PAGE	15	
0381		FUNCT	ION} lin			
O3B1	3A 3FF2	LFEED:	LDA	AULF	;auto line feed?	
03B4	B7		ORA	A		
03B5 03B8	C2 03C1 C9		JNZ RET	LFD	;do line feed	
0389	3A 3FF2	; {FUNCT CARRTN:		riage re AULF	turn ;auto line feed?	
OBBC	B7	CHARTER.	ORA	A		
O3BD	C2 04DE		JNZ	ZROCUR	ido cr only	
0300	47	LSTCHR:	MOV	В, А	;set cursor to 1st col	
0301	11 3FFC	LFD:	LXI	D, LROW		
0304	2A 3FFA		LHLD	CROW		
0307 0308	24 1A		INR LDAX	H D		
0309	BD		CMP	Ľ	;crow=lrow?	
03CA	7E		MOV	A, M	inext row	
OGCB OGCE	32 3FFA 32 3FEB	DICMD:	STA STA	CROW ICMD	;crow+1 ;defeat insert char mode	
03D1	со		RNZ		;not last row	
0302	EB	LFSCR:	XCHG		;last row, do scroll	
03D3	F3	2. 2011.	DI			
03D4 03D5	77 2E FE		MOV MVI	M, A L, low Fi	; 1row+1 Row	
03D7	5E		MOV	E, M	;de≖row wrap arround tbl	
0308	1A		LDAX	D		
03D9 03DA	77 FB		MOV EI	M, A	; frow+1	
		FUNCT	ION} cle	ar curso	г гом	
OBDB	2A 3FFA		LHLD	CROW		
OBDE	EB	CLRROW:	хоне			
OBDE	21 0000	CLARUW.	LXI	н, о		
03E2	39		DAD XCHG	SP		
03E3 03E4	EB F3		DI			
03E5	F9		SPHL			
03E6 03E7	E1 F9		POP SPHL	н	;lookup row start ;sp≖row start	
03E8	FB		EI			
03E9	2A 3F20	PAGE	LHLD	FBG	;space + attribute	
CTADDI EV	MACRO-ASSEMBL			PAGE	16	
CRTB01	MACKU-ASSCHIDE	ER VZ. U		FAGE	18	
OBEC						
03EC 03ED	E5 E5	PUSHSP:	PUSH PUSH	н н	;do clear row	
OJEE	E5		PUSH	н		
03EF 03F0	E5 E5		PUSH PUSH	н н		
03F0	E5		PUSH	н		
03F2 03F3	E5 E5		PUSH PUSH	H H		
03F3 03F4	E5 E5		PUSH	н		
03F5	E5		PUSH	н	; 10	
03F6 03F7	E5 E5		PUSH PUSH	H H		
03F8	E5		PUSH	н		
03F9 03FA	E5 E5		PUSH PUSH	н н		
O3FB	E5		PUSH	н		
O3FC O3FD	E5 E5		PUSH PUSH	н н		
03FE	E5		PUSH	н		
03FF 0400	E5 E5		PUSH PUSH	н н	; 20	
0400	E5		PUSH	H		
0402	E5		PUSH	н		
0403 0404	E5 E5		PUSH PUSH	н н		
0405	E5		PUSH	н		
0406 0407	E5 E5		PUSH PUSH	н н		
0408	E5		PUSH	н		
0409 040A	E5 E5		PUSH PUSH	н н	; 30	
040B	E5		PUSH	н		
0400	F3		DI			TL/F/5869-16
						.2/1/0000-10

040D	E5		PUSH	н		
040E	E5		PUSH	н		
040F	E5		PUSH	н		
0410	E5		PUSH	н		
0411	E5		PUSH	н		
0412	E5		PUSH	н		
0413	E5		PUSH	н	••	
0414	E5		PUSH	н	; 40	
0415	EB		XCHG SPHL			
0416	F9 FB	BSRTN:	EI			
0417	C9	DONIN.	RET			
0418	07		1194			
		; {FUNCT	ION} dela	ete		
0419	3E 20	DEL:	MVI	A, SPC		
041B	CD 06CB		CALL	STSP	;store space	
041E	AF		XRA	A	;do back space	
		PAGE				
STARPLE	MACRO-ASSEMBL	ER V2.0		PAGE	17	
CRTB01						
041F						
0411		; (FUNCT	ION} bac	k space		
041F	C4 06C1	BS:		CHKGM	;check graphics mode	
0422	F3	BS1:	DI			
0423	05		DCR	В	; cursor-1	
0424	F2 0417		JP	BSRTN	;not ist column	
0427	04		INR	в		
0428	FB		EI			
0429	3A 3FEF		LDA	GECNTL		
0420	B7		ORA	Α	graphics mode?	
042D	co		RNZ		;defeat wrap arround	
042E	CD 0451		CALL	UCUR1	; up cursor one row	
0431	C8		RZ	B 70	; crow=frow?	
0432 0434	06 4F D3 01	BELL:	MVI OUT	B,79 BELPRT	;set cursor to last col	
0434	C9		RET	DELFRI		
0438	67		NC I			
		FUNCT	ION} for	ward cure	507	
0437	CD 06C1	FS:	CALL		icheck graphics mode	
043A	3E 4F		MVI	A, 79	····· ··· ··· ····	
0430	BB		CMP	в	;last column?	
0430	C2 0387		JNZ	MIDCHR	;not last column	
0440	3A 3FEF		LDA	GECNTL		
0443	B7		ORA	A	;graphics mode?	
0444	CO		RNZ		;defeat wrap arround	
0445	CD 0463		CALL	DCUR1	idown cursor one row	
0448	C8		RZ		;crow=lrow?	
0449	06 00		MVI	B, O	;set cursor to 1st col	
044B	D3 01		OUT	BELPRT		
044D	C9		RET			
			ION) up (UTSOT OF	De Fow	
044F	CD 06C1		CALL	CHKGM)cneck graphics mode	
044E 0451	CD 06C1 11 3FFA	UPCUR: UCUR1:		CHKGM D, CROW	;check graphics mode	
		UPCUR:			;cneck grapnics mode	
0451	11 3FFA	UPCUR:	LXI	D, CROW	; crow	
0451 0454 0457 0458	11 3FFA 2A 3FFE 1A 24	UPCUR:	LXI LHLD LDAX INR	D, CROW FROW D H	;crow ;hl=row wrap arround tbl	
0451 0454 0457 0458 0459	11 3FFA 2A 3FFE 1A 24 BD	UPCUR:	LXI LHLD LDAX INR CMP	D,CROW FROW D H L	; crow	
0451 0454 0457 0458 0459 0459	11 3FFA 2A 3FFE 1A 24 BD C2 04A7	UPCUR:	LXI LHLD LDAX INR CMP JNZ	D,CROW FROW D H L \$4	;crow ;hl≖row wrap arround tbl ;crow=frow?	
0451 0454 0457 0458 0459 0459 0450	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01	UPCUR:	LXI LHLD LDAX INR CMP JNZ DUT	D,CROW FROW D H L \$4	;crow ;hl=row wrap arround tbl	
0451 0454 0457 0458 0459 0459	11 3FFA 2A 3FFE 1A 24 BD C2 04A7	UPCUR:	LXI LHLD LDAX INR CMP JNZ	D,CROW FROW D H L \$4	;crow ;hl≖row wrap arround tbl ;crow=frow?	
0451 0454 0457 0458 0459 0459 0450	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01	UPCUR: UCUR1:	LXI LHLD LDAX INR CMP JNZ DUT RET	D, CROW FROW D H L \$4 BELPRT	;crow ;hl≖row wrap arround tbl ;crow=frow? ;crow=frow, ring bell	
0451 0454 0457 0458 0459 0459 045A 045D 045F	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9	UPCUR: UCUR1:	LXI LHLD LDAX INR CMP JNZ DUT RET ION} down	D, CROW FROW D H L \$4 BELPRT	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row	
0451 0457 0457 0458 0459 045A 045A 045A 045F 045F	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1	UPCUR: UCUR1: ; {FUNCT DWNCUR:	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down	D, CROW FROW D H L \$4 BELPRT CHKGM	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell	
0451 0454 0457 0458 0459 0459 0454 0450 045F 0455	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA	UPCUR: UCUR1:	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row	
0451 0457 0457 0459 0459 0459 0450 045F 045F 0460 0463	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC	UPCUR: UCUR1: ; {FUNCT DWNCUR:	LXI LHLD LDAX INR CMP JNZ GUT RET ION} down CALL LXI LXI LHLD	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW LROW	;crow ;hl≖row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode	
0451 0457 0457 0458 0459 045A 045A 045A 045F 0460 0463 0460 0463	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A	UPCUR: UCUR1: ; {FUNCT DWNCUR:	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LDAX	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW LROW D	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow	
0451 0457 0457 0459 0459 0459 0450 045F 045F 0460 0463	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC	UPCUR: UCUR1: ; {FUNCT DWNCUR:	LXI LHLD LDAX INR CMP JNZ GUT RET ION} down CALL LXI LXI LHLD	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW LROW	;crow ;hl≖row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode	
0451 0454 0457 0458 0459 0459 0459 0455 0455 0455 0460 0463 0466 0469 0466	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 24	UPCUR: UCUR1: ; {FUNCT DWNCUR:	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LDAX INR	D, CROW FROW D H L BELPRT CHKGM D, CROW LROW D H	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl	
0451 0457 0457 0458 0459 045A 045A 045F 0460 0463 0466 0464 0468	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 24 BD	UPCUR: UCUR1: ; {FUNCT DWNCUR:	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LDAX INR CMP	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW LROW D H L S \$5	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl	
0451 0457 0458 0459 0459 045A 045D 045F 045F 0460 0463 0463 0465 0469 0468 0468 0468	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 24 BD C2 04A9	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1:	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LHLD LDAX INR CMP JNZ	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW LROW D H L S \$5	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow?	
0451 0457 0457 0459 0459 0450 0450 0450 045F 0463 0463 0464 0469 0464 0469 0464 0465	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 2A 3FFC 1A 24 BD C2 04A9 D3 01	UPCUR: UCUR1: ; {FUNCT DWNCUR:	LXI LHLD LDAX INR CMP JNZ UUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ UUT	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW LROW D H L S \$5	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow?	
0451 0457 0457 0458 0459 045A 045A 045A 045F 0460 0463 0464 0463 0464 0465 0464 0468 0465 0467 0467	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 2A 3FFC 1A 2D C2 04A9 D3 01 C9	UPCUR: UCUR1: ; (FUNCT DUNCUR: DCUR1: PAGE	LXI LHLD LDAX INR CMP JNZ UUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ UUT	D, CROW FROW D H SELPRT D, CUTSOF CHKGM D, CROW LROW D H L S BELPRT	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell	
0451 0457 0457 0458 0459 045A 045A 045F 045F 0460 0463 0464 0469 0464 0469 0464 0465 0467 0465 0467 0465 0465 0465 0465 0465 0465 0465 0465	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 2A 3FFC 1A 24 BD C2 04A9 D3 01	UPCUR: UCUR1: ; (FUNCT DUNCUR: DCUR1: PAGE	LXI LHLD LDAX INR CMP JNZ UUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ UUT	D, CROW FROW D H L \$4 BELPRT CHKGM D, CROW LROW D H L S \$5	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow?	
0451 0457 0457 0459 0459 0450 0450 045F 0460 0463 0466 0465 0466 0466 0466 0467 0466 0467 0466 0465 0471 STARPLE) CRT801	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 2A 3FFC 1A 2D C2 04A9 D3 01 C9	UPCUR: UCUR1: ; (FUNCT DUNCUR: DCUR1: PAGE	LXI LHLD LDAX INR CMP JNZ UUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ UUT	D, CROW FROW D H SELPRT D, CUTSOF CHKGM D, CROW LROW D H L S BELPRT	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell	
0451 0457 0457 0458 0459 045A 045A 045F 045F 0460 0463 0464 0469 0464 0469 0464 0465 0467 0465 0467 0465 0465 0465 0465 0465 0465 0465 0465	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 2A 3FFC 1A 2D C2 04A9 D3 01 C9	UPCUR: UCUR1: ; (FUNCT DUNCUR: DCUR1: PAGE ER V2. 0	LXI LHLD LDAX INR CMP JNZ DUT RET ION} down CALL LXI LHLD LHLD LHLD LHLD LHLD NR CMP JNZ OUT RET	D, CROW FROW D H H SELPRT D, CUTSOF CHKGM D, CROW LROW D H H L \$5 BELPRT PAGE	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell	
0451 0457 0457 0458 0459 045A 045A 045C 0460 0463 0464 0465 0465 0466 0467 0466 0467 0467 0467 0471	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 24 BD C2 04A9 D3 01 C9 C2 04A9 D3 01 C9	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ OUT RET	D, CROW FROW D H L \$4 BELPRT CUTSOF CHKGM D, CROW LROW D H L S BELPRT PAGE	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell	
0451 0457 0457 0459 0459 0450 0450 045F 0460 0463 0464 0464 0465 0466 0465 0466 0465 0466 0467 0471 STARPLE) CRT801 0472	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 2A 3FFC 1A 2A 3FFC 1A 2A 3FFC 1A 2A 3FFC 1A 2A 3FFC 1A 2A 3FFE 13 7FE 14 24 20 20 20 20 20 20 20 20 20 20 20 20 20	UPCUR: UCUR1: ; (FUNCT DUNCUR: DCUR1: PAGE ER V2. 0	LXI LHLD LDAX INR CMP JNZ DUT RET ION} down CALL LXI INR CMP JNZ CMP JNZ TNR CMP JNZ INN RET	D, CROW FROW D H H SELPRT CHKGM D, CROW LROW D H L S S BELPRT PAGE	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell	
0451 0457 0457 0458 0459 045A 045A 045A 045F 045F 0460 0463 0466 0465 0466 0469 0466 0468 0466 0468 0466 0468 0466 0471 0472 0472 0472 0475	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 24 BD C2 04A9 D3 01 C9 C2 04A9 D3 01 C9 C9 C1 C9 C2 04A9 D3 01 C9 C1 C9 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ OUT RET ION} rol LXI LHLD	D, CROW FROW D H L \$4 BELPRT D, CROW D, CROW EL \$5 BELPRT PAGE	;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell	
0451 0457 0457 0459 0459 0450 0450 0450 0450 0460 0463 0466 0465 0466 0466 0467 0466 0467 0466 0467 0471 0472 0472 0472 0475	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A C2 04A9 D3 01 C9 C9 C2 04A9 D3 01 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ DUT RET ION} down CALL LXI INR CMP JNZ CMP JNZ TNR CMP JNZ INN RET	D, CROW FROW D H H SELPRT CHKGM D, CROW LROW D H L S S BELPRT PAGE	; crow ; h1=row wrap arround tb1 ; crow=frow? ; crow=frow, ring bell one row ; check graphics mode ; crow ; h1=row wrap arroud tb1 ; crow=1row? ; crow=lrow, ring bell 18	
0451 0457 0457 0458 0459 045A 045A 045F 0466 0463 0466 0466 0466 0466 0466 0466	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 2A 3FFC 1A C9 C1 C0 C9 C1 C9 C1 C9 C1 C9 C1 C9 C1 C9 C1 C9 C1 C9 C1 C9 C1 C1 C9 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1 C1	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LDAX INR RET ION} tol LXI LHLD LXI LXI LHLD LXI LHLD LDAX INR	D, CROW FROW D H H EL SELPRT CHKGM D, CROW L LROW D H L LROW D PAGE	; crow ; hl=row wrap arround tb1 ; crow=frow? ; crow=frow, ring bell one row ; check graphics mode ; crow ; hl=row wrap arroud tb1 18 ; hl=row wrap arround tb1	
0451 0457 0457 0459 0459 0450 0450 0450 0450 0460 0463 0466 0465 0466 0466 0467 0466 0467 0466 0467 0471 0472 0472 0472 0475	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A C2 04A9 D3 01 C9 C9 C2 04A9 D3 01 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ OUT CALL LXI LHLD LDAX INR CMP JNZ OUT RET ION} rol LXI LHLD LDAX	D, CROW FROW D H H SELPRT D, CROW D, CROW L KOW D H H CROW D H CROW D H CROW D H FROW D H H	; crow ; h1=row wrap arround tb1 ; crow=frow? ; crow=frow, ring bell one row ; check graphics mode ; crow ; h1=row wrap arroud tb1 ; crow=1row? ; crow=lrow, ring bell 18	
0451 0457 0457 0458 0459 0459 045A 045A 045F 045F 0460 0463 0465 0465 0466 0466 0466 0466 0466 0466	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 24 BD C2 04A9 D3 01 C9 C2 04A9 D3 01 C9 C9 C1 3FFA 2A 3FFE 1A 2A 3FFE 1A	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ DUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ OUT RET ION} rol LXI LHLD LDAX INR CMP MOV DI	D, CROW FROW D H H BELPRT D, CROW CHKGM D, CROW CHKGM D, CROW C BELPRT PAGE I up D, CROW D, CROW FROW D H L A, M	<pre>;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell 18 ;hl=row wrap arround tbl ;crow=frow? ;next row</pre>	
0451 0457 0457 0459 0459 0459 0450 0450 0450 0463 0463 0463 0464 0465 0465 0466 0466 0467 0467 0471 0471 0472 0472 0475 0475 0475 0475 0475 0475 0475 0475	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 C0 06C1 11 3FFA 2A 3FFC 1A 24 BD C2 04A9 D3 01 C9 C9 C9 C1 C9 C1 C9 C1 C9 C1 C1 C9 C2 04A7 C2 04A9 C2 04A9 C2 04A9 C3 C2 04A9 C3 C4 C9 C3 C4 C9 C4 C4 C5 C9 C4 C4 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5 C5	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ OUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ OUT RET ION} rol LXI LHLD LDAX INR CMP MOV DI STA	D, CROW FROW D H H EL SELPRT CHKGM D, CCOW LROW D H L LROW D H S BELPRT PAGE	<pre>;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell 18 ;hl=row wrap arround tbl ;crow=frow?</pre>	
0451 0457 0457 0458 0459 045A 045A 045A 045A 045F 045F 0466 0469 0469 0469 0466 0469 0466 0467 0468 0467 0471 0472 0472 0472 0475 0478 0479 0478 0478	11 3FFA 2A 3FFE 1A 24 BD C2 04A7 D3 01 C9 CD 06C1 11 3FFA 2A 3FFC 1A 24 BD C2 04A9 D3 01 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9 C9	UPCUR: UCUR1: ; (FUNCT DWNCUR: DCUR1: PAGE ER V2.0 ; (FUNCT	LXI LHLD LDAX INR CMP JNZ DUT RET ION} down CALL LXI LHLD LDAX INR CMP JNZ OUT RET ION} rol LXI LHLD LDAX INR CMP MOV DI	D, CROW FROW D H H BELPRT D, CROW CHKGM D, CROW CHKGM D, CROW C BELPRT PAGE I up D, CROW D, CROW FROW D H L A, M	<pre>;crow ;hl=row wrap arround tbl ;crow=frow? ;crow=frow, ring bell one row ;check graphics mode ;crow ;hl=row wrap arroud tbl ;crow=lrow? ;crow=lrow, ring bell 18 ;hl=row wrap arround tbl ;crow=frow? ;next row</pre>	TL/F/5869-17

0483 0484	12 EB	RUNEQ	STAX XCHG	D	; crow+1 ; d=1, h=3Fh	
0485	2E FC	Noned.	MVI	L, low L		
0487	5E		MOV	E, M	;de≖row wrap arround tbl	
0488 0489	1A 77		LDAX MOV	D M, A	;next row ;lrow+1	
048A	FB		EI	10.6	/1/00.1	
048B	C9		RET			
0480	11 3FFA	; (FUNC) ROLDWN:	10N} ro	11 down D/CROW		
048F	3A 3FFC	NOLDWIN.	LDA	LROW		
0492	D6 04		SVI	4	;up one row	
0494	6F		MOV	L, A		
0495 0497	26 01 1A		MVI LDAX	H, 1 D	;hl≕row wrap arround tbl ;crow	
0498	D6 04		SUI	4) (100	
049A	BD		CMP	L	;crow=lrow?	
049B	7E		MOV	A, M	;up one row	
049C 049D	F3 32 3FFC		DI STA	LROW	; lrow-1	
0440	C2 04A4		JNZ	RDNEG	,1100-1	
04A3	12		STAX	D	; crow-1	
0444	1E FE	RDNEG	MVI	E,low F	ROW	
0446	1A	* 4	LDAX	D		
04A7 04A9	D6 04 6F	\$4: \$5:	SUI MOV	4 L, A	;hl=row wrap arround tbl	
0444	7E	4 0.	MOV	A, M	hit-low wish alloons cor	
04AB	12		STAX	D	; frow-1	
04AC	FB		EI			
04AD	C9		RET			
04AE	AF	; (FUNCT CURULK:	ION} en	able curs A	sor, unlock keyboard	
04AF	32 3FEE	CONVER	STA		;unlock keyboard	
04B2	2F		CMA		;enable cursor	
04B3	C3 001B		JMP	DICUR		
		: (FUNCT	10N} 1o	ck keyboa	ard	
0486	32 3FEE	KBLK:	STA		ilock keyboard	
0489	C9	-	RET			
		PAGE				
	K MACRO-ASSE	MBLER V2.0		PAGE	19	
CRTB01						
04BA		: (FUNCT	TON: de	lete rest	of name	
04BA	3A 3FFA		LDA	CROW	, of page	
04BD	CD 04E9		CALL	CTLRW2	;clear crow+1 to lrow	
		(5)(0)	TON'S da	lete rest	of line	
			10113 06.			
0400	CD 0171	DRTLN:	CALL	DFCLOC	iget cursor loc and ditt	
04C0 04C3	CD 0171 57		CALL MOV	DFCLOC D, A	;get cursor loc and diff ;save	
04C3 04C4	57 3A 3F20	DRTLN:	MOV LDA	D, A FBG	; save	
04C3 04C4 04C7	57 3A 3F20 77		MOV LDA MOV	D, A FBG M, A		
04C3 04C4 04C7 04C8	57 3A 3F20 77 23	DRTLN:	MOV LDA MOV INX	D,A FBG M,A H	; save	
04C3 04C4 04C7 04C8 04C9	57 3A 3F20 77 23 15	DRTLN:	MOV LDA MOV INX DCR	D,A FBG M,A H D	;save ;store space/attribute	
04C3 04C4 04C7 04C8	57 3A 3F20 77 23	DRTLN:	MOV LDA MOV INX	D,A FBG M,A H	; save	
04C3 04C4 04C7 04C8 04C9 04C9	57 3A 3F20 77 23 15 C2 04C7	DRTLN: DRLLP:	MOV LDA MOV INX DCR JNZ RET	D, A FBG M, A H D DRLLP	;save ;store space/attribute ;until end of line	
04C3 04C4 04C7 04C8 04C9 04C9 04CA 04CD	57 3A 3F20 77 23 15 C2 04C7 C9	DRTLN: DRLLP: ; (FUNCT	MOV LDA MOV INX DCR JNZ RET	D, A FBG M, A H D DRLLP ear scree	;save ;store space/attribute ;until end of line :n	
04C3 04C4 04C7 04C8 04C9 04C9	57 3A 3F20 77 23 15 C2 04C7	DRTLN: DRLLP:	MOV LDA MOV INX DCR JNZ RET	D, A FBG M, A H D DRLLP ear scree	;save ;store space/attribute ;until end of line	
04C3 04C4 04C7 04C8 04C9 04C9 04C0 04CD 04CE 04D1 04D4	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D	DRTLN: DRLLP: ; (FUNCT	MOV LDA MOV INX DCR JNZ RET CALL LHLD MOV	D, A FBG M, A H D DRLLP DEGRPH FROW A, L	;save ;store space/attribute ;until end of line m ;defeat graphics mode	
04C3 04C4 04C7 04C8 04C9 04CA 04CD 04CD	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE	DRTLN: DRLLP: ; (FUNCT	MOV LDA MOV INX DCR JNZ RET TION} cli CALL LHLD	D, A FBG M, A H D DRLLP DEGRPH FROW A, L	;save ;store space/attribute ;until end of line :n	
04C3 04C4 04C7 04C8 04C9 04C9 04C0 04CD 04CE 04D1 04D4	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D	DRTLN: DRLLP: ; (FUNC) CLRSCN:	MOV LDA MOV INX DCR JNZ RET CALL LHLD MOV CALL	D, A FBG M, A H D DRLLP DEGRPH FROW A, L	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow	
04C3 04C4 04C7 04C8 04C9 04C4 04CD 04CD 04CE 04D1 04D4 04D5 04D5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE	DRTLN: DRLLP: ; (FUNC1 CLRSCN: ; (FUNC1	MOV LDA MOV INX DCR JNZ RET CALL LHLD MOV CALL ION) hor LDA	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 ne cursor FROW	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow	
04C3 04C4 04C7 04C8 04C9 04C6 04CD 04CE 04D1 04D4 04D5 04D5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR:	MOV LDA MOV INX DCR JNZ RET TION} cli CALL LHLD MOV CALL CALL TION> hor LDA STA	D.A FBG M.A H D DRLLP ERCW A,L CTLRW1 me cursor FRDW CRDW	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow	
04C3 04C4 04C7 04C8 04C9 04CA 04C4 04C4 04C4 04C4 04D1 04D4 04D5 04D5 04D8 04D8 04D8 04D8	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00	DRTLN: DRLLP: ; (FUNC1 CLRSCN: ; (FUNC1	MOV LDA MOV INX JNZ RET ION} cli CALL LHLD MOV CALL ION} hor STA MVI	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 ne cursor FROW	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow	
04C3 04C4 04C7 04C8 04C9 04C6 04CD 04CE 04D1 04D4 04D5 04D5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR:	MOV LDA MOV INX DCR JNZ RET TION} cli CALL LHLD MOV CALL CALL TION> hor LDA STA	D.A FBG M.A H D DRLLP ERCW A,L CTLRW1 me cursor FRDW CRDW	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow	
04C3 04C4 04C7 04C8 04C9 04CA 04CD 04CD 04CE 04D1 04D4 04D5 04D5 04D8 04D8 04D8 04D8 04D8	57 3A 3F20 77 23 15 62 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR	MOV LDA MOV INX DCR JNZ RET CALL LHLD MOV CALL LHLD MOV CALL LDA STA MVI RET TO LAST	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 TROW CROW B.O	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow	
04C3 04C4 04C7 04C8 04C9 04C4 04CD 04CE 04D1 04D4 04D5 04D5 04D8 04D8 04D8 04D8 04D8 04D8 04D8	57 3A 3F20 77 23 15 C2 04C7 C7 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR:	MOV LDA MOV INX DCR JNZ RET TON} cli CALL LHLD MOV CALL LHLD MOV CALL LHA MVI RET TO LAST MVI	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 TROW CROW B.O ROW D.1	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to 1st col	
04C3 04C4 04C7 04C8 04C9 04CA 04CD 04CD 04CE 04D1 04D4 04D5 04D5 04D8 04D8 04D8 04D8 04D8	57 3A 3F20 77 23 15 62 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR	MOV LDA MOV INX DCR JNZ RET CALL LHLD MOV CALL LHLD MOV CALL LDA STA MVI RET TO LAST	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 CTLRW1 Me cursor FROW CROW B.O ROW D.1 E.A	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to 1st col ;de=row wrap arround tbl	
04C3 04C4 04C7 04C8 04C9 04C4 04CD 04CE 04D1 04D4 04D5 04D5 04D5 04D8 04D8 04D8 04D8 04D8 04E0 04E1 04E3 04E4 04E5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6F	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR CLRWLP:	MOV LDA MOV INX DCR JNZ RET TON} cli CALL HLD MOV CALL ION) hor LDA STA MVI RET TO LAST MVI MOV LDAST MVI MOV	D.A FBG M.A H D DRLLP PRLLP PRDW A.L CTLRW1 MCCUTSOT FRDW CROW B.O ROW D.1 E.A D L.A	<pre>;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to lst col ;de=row wrap arround tbl ;next row</pre>	
04C3 04C4 04C7 04C8 04C9 04C4 04C9 04C4 04C1 04C1 04D5 04D5 04D5 04D5 04D5 04D5 04D5 04D5	57 3A 3F20 77 23 15 62 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6F CD 03DE	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR CLRWP:	MOV LDA MOV INX DCR JNZ RET ION} cli CALL LHLD MOV CALL TO LAST MVI RET TO LAST MOV LDAX MOV LDAX CALL	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 ROW B.O ROW D.1 E.A D L.A CLRROW	;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to 1st col ;de=row wrap arround tbl	
04C3 04C4 04C7 04C8 04C9 04CD 04CD 04CD 04CD 04CD 04D5 04D5 04D5 04D5 04D5 04D5 04D5 04E1 04E3 04E1 04E3 04E4 04E5 04E5	57 3A 3F20 77 23 15 C2 04C7 C7 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6F CD 03DE 2A 3FFC	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR CLRWLP:	MOV LDA MOV INX DCR JNZ RET TON3 cl. CALL LHLD MOV CALL LDA STA MVI RET TO LAST MVI MOV LDAST MOV CALL LHLD	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 TROW CROW D.1 E.A D L.A CLRROW LROW	<pre>;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to lst col ;de=row wrap arround tbl ;next row ;clear whole row</pre>	
04C3 04C4 04C7 04C8 04C9 04C6 04CD 04CE 04D1 04D4 04D5 04D5 04D8 04D8 04D8 04D8 04D8 04D8 04D8 04E0 04E1 04E3 04E4 04E5 04E4 04E5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6F CD 03DE 2A 3FFC BD	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR CLRWP:	MOV LDA MOV INX DCR JNZ RET TION} cli CALL HLD MOV CALL LDA STA MVI RET TO LAST MVI MOV LDAST MVI MOV CALL LHLO CALL LHLO CALL LHLO CALL CAST	D.A FBG M.A H DRLLP PRLLP PRDW A.L CTLRW1 CTLRW1 MCCOW B.O ROW D.1 E.A D.1 D.1 E.A D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1	<pre>;save ;store space/attribute ;until end of line ; defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to lst col ;de=row wrap arround tbl ;next row ;clear whole row ;row=lrow?</pre>	
04C3 04C4 04C7 04C8 04C9 04CD 04CD 04CD 04CD 04CD 04D5 04D5 04D5 04D5 04D5 04D5 04D5 04E1 04E3 04E1 04E3 04E4 04E5 04E5	57 3A 3F20 77 23 15 C2 04C7 C7 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6F CD 03DE 2A 3FFC	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR CLRWP:	MOV LDA MOV INX DCR JNZ RET TON3 cl. CALL LHLD MOV CALL LDA STA MVI RET TO LAST MVI MOV LDAST MOV CALL LHLD	D.A FBG M.A H DRLLP PRLLP PRDW A.L CTLRW1 CTLRW1 MCCOW B.O ROW D.1 E.A D.1 D.1 E.A D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1	<pre>;save ;store space/attribute ;until end of line en ;defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to lst col ;de=row wrap arround tbl ;next row ;clear whole row</pre>	
04C3 04C4 04C7 04C8 04C9 04CD 04CD 04C2 04C1 04D5 04D5 04D5 04D5 04D5 04D5 04D5 04D5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6D 03DE 2A 3FFC BD C2 04E1	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ; CLEAR CLRWP:	MOV LDA MOV JNZ RET JNZ RET CALL CALL LHLD CALL LDA STA MOV CALL LDAX MOV LDAX MOV LDAX MOV LDAX MOV CALL LDAX MOV CALL LDAX MOV CALL LDAX MOV	D.A FBG M.A H DRLLP PRLLP PRDW A.L CTLRW1 CTLRW1 MCCOW B.O ROW D.1 E.A D.1 D.1 E.A D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1 D.1	<pre>;save ;store space/attribute ;until end of line ; defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to lst col ;de=row wrap arround tbl ;next row ;clear whole row ;row=lrow?</pre>	
04C3 04C4 04C7 04C8 04C9 04C4 04CD 04CE 04D1 04D4 04D5 04D5 04D8 04D8 04D5 04D8 04D8 04D5 04E1 04E3 04E4 04E5 04E4 04E5 04E5 04E5 04E5 04E5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6F CD 03DE 2A 3FFC BD C2 04E1 C9	DRTLN: DRLLP: ; (FUNCT CLRSCN: ; (FUNCT HOMCUR: ZROCUR: ZROCUR: ; CLEAR CLRWLP: CTLRW1: CTLRW2: ; (FUNCT	MOV LDA MOV LDA MOV INX JNZ RET TON} cli CALL HLD MOV CALL LDA STA MVI RET TO LAST MVI CALL LDAY RET TO LAST MVI CALL LHLD UDA RET TON JNZ RET	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 CTLRW1 CTLRW1 CCLRW1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 A D.1 A D.1 E.A D.1 E.A D.1 E.A D.1 E.A D.1 E.A D.1 A D.1 E.A D.1 A A B.1 A A D.1 A D.1 A D.1 A D.1 A D.1 A D.1 A C A D.1 A D.1 C A D.1 C A D.1 A C A D.1 E.A D.1 E.A D.1 A C C A D.1 C C C C C C C C C C C C C C C C C C C	<pre>;save ;store space/attribute ;until end of line ; defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to lst col ;de=row wrap arround tbl ;next row ;clear whole row ;row=lrow? ;until last row follows</pre>	
04C3 04C4 04C7 04C8 04C9 04CD 04CD 04C2 04C1 04D5 04D5 04D5 04D5 04D5 04D5 04D5 04D5	57 3A 3F20 77 23 15 C2 04C7 C9 CD 06BC 2A 3FFE 7D CD 04E6 3A 3FFE 32 3FFA 06 00 C9 16 01 5F 1A 6D 03DE 2A 3FFC BD C2 04E1	DRTLN: DRLLP: ; (FUNCT CLRSCN: ZROCUR: ; CLEAR CLRWLP: CTLRW1: CTLRW1:	MOV LDA MOV LDA MOV INX JNZ RET TON} cli CALL HLD MOV CALL LDA STA MVI RET TO LAST MVI CALL LDAY RET TO LAST MVI CALL LHLD UDA RET TON JNZ RET	D.A FBG M.A H D DRLLP ear scree DEGRPH FROW A.L CTLRW1 CTLRW1 CTLRW1 CCLRW1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 D.1 E.A D.1 A D.1 A D.1 E.A D.1 E.A D.1 E.A D.1 E.A D.1 E.A D.1 A D.1 E.A D.1 A A B.1 A A D.1 A D.1 A D.1 A D.1 A D.1 A D.1 A C A D.1 A D.1 C A D.1 C A D.1 A C A D.1 E.A D.1 E.A D.1 A C C A D.1 C C C C C C C C C C C C C C C C C C C	<pre>;save ;store space/attribute ;until end of line ; ;defeat graphics mode ;clear frow to lrow ;crow=frow ;set cursor to lst col ;de=row wrap arround tbl ;next row ;clear whole row ;row=lrow? ;until last row</pre>	

0457				ckground		
04F7 04FA	11 A0A0 21 0000	BGNDF: LDFGD:	LXI LXI	D, 0A0A0 H, 0	h;background spaces	
04FD	39	LDFGD	DAD	SP		
04FE	EB		XCHG	0.		
04FF	31 3F50		LXI	SP, DMYR	0₩+80	
0502	C3 OBEC		JMP		;store in dummy row	
		PAGE				
STARPLEY	MACRD~ASSEM	BIER V2 0		PAGE	20	
CRT801						
0505						
		; {FUNCT	ION} in	sert a li	ne	
0505	CD 0590	INSLNE:		CRLR	;calc crow/lrow diff	
0508	21 3FFA		LXI	H, CROW		
0508	73		MOV	M, E	;set crow=lrow	
050C 050F	C2 054B C3 058C		JNZ JMP	MOVDWN DRWZCU	;move row contents ;else, del lrow/zero cur	
0000	00 0000		UT //	DINALOO	ferse, der from/refo con	
		; (FUNCT	ION} st	rip off a	line	
0512	CD 0590	STOFLNE	CALL	CRLR	;get crow∕lrow diff	
0515	CA 058C		JZ	DRWZCU	;del lrow∕zero cursor	
0518	E5	SLNERG:		H	;else do move	
0519	21 3FE7			H, CPYCT		
051C 051E	36 00 CD 0550		MVI CALL	M,O MOVROW	;copy upward ;move row contents	
0521	E1 0550		POP	HUVRUW	;move row contents ;get original crow	
0522	22 3FFA		SHLD	CROW	iback to crow	
0525	C9		RET		-	
					p off line with range	
0526	2A 3FFA	ISLRG	LHLD	CROW		
0529	3A 3FA3				;read 2nd parameter	
052C 052D	3D E6 3F		DCR ANI	A O3Fh	;40/7Fh offset to 0/3Fh	
052F	FE 38		CPI	038h	, to man offset to orden	
0531	DO		RNC	00011	; error	
0532	FE 17		CPI	017h		
0534	DA 0530		JC	ISNPA		
0537	FE 20		CPI	020h		
0539	DB		RC	_	; error	
053A	D6 09	TONDA	SUI	9		
0530	3C 57	ISNPA:	INR MOV	А D, A		
053D 053E	3A 3FA4		LDA	LINP+1	;read 1st parameter	
0541	FE 53		CPI	"S"	strip off?	
0543	7A		MOV	A, D	· · · · · ·	
0544	CA 0518		JZ	SLNERG	;do strip off line	
0547	CD 05C1	ILNERG:		IRWOS	;offset row by para	
054A	7A		MOV	A, D	;return para	
054B 054E	21 3FE7 36 04	MOVDWN:	LXI MVI	H, CPYCT		
0548	36 04		mv1	M, 4	;copy downward	
		MOVE R	OW CONT	ENTS, UP/D	OWN CNTL BY CPYCTL	
0550	F5	MOVROW:		PSW	isave row count	
0551	2A 3FFA		LHLD	CROW		
0554	5E		MOV	E, M		
0555	23		INX	H L		
0556	56		MOV	D, M	· · · · · · · · · · · · · · · · · · ·	
0557	1B D5		DCX	ם ם	icrow end	
0558 0559	D5 21 3FE7		PUSH LXI		;save L;direction control	
0550	11 3FFA			D, CROW	2,91,65010H 20H0F01	
055F	14		LDAX	D	;read crow	
0560	96		SUB	M	direction cntl	
		PAGE				
	MACRO-ASSEM			PAGE	21	
CRT801	HACKU-ASSEM	DEER VZ. U		FAGE	£. 4	
0561						
0561	6F		MOV	L, A		
0562	26 01		MVI	H, 1	;hl≖row wrap arround tbl	
0564	7E		MOV	A, M	;lookup +/- one row	
0565	12		STAX	D	iupdate crow	
0566	6F		MOV	LA		
0567	25		DCR	H L	;hl≖row start table	
0568	5E		MOV	E, M		
0569 056A	23 56		INX MOV	н D, M		
056B	18		DCX	D, M D	;+/- row end	
0560	E1		POP	H	;+/- row end ;rtn current row last loc	
056D	06 10		MVI	B,80/5	copy 80 characters	
056F	1A	CPLP:	LDAX	D	read	
0570	77		MOV	M, A	; сору	
0571	2B		DCX	н	;next byte	
0572	1B		DCX	D	inext byte	
0573 0574	1A 77		LDAX MOV	D M,A	;do 5 times for speed	

0575	28			DCX	н		
0576	18			DCX LDAX	ם ם	; 3	
0577 0578	14			MOV	M, A	, .	
0575	21			DCX	н		
0574	1 E			DCX	D		
0578	14			LDAX	D	; 4	
0570	77			MOV	M, A		
057E 057E	2E 1E			DCX DCX	H D		
0576	14			LDAX	D	; 5	
0580	77			MOV	м, A	, 0	
0581	21			DCX	н		
0582	1 E			DCX	D		
0583	05			DCR	B		
0584		2 056F				;finish 80 bytes?	
0587	FI			P O P D C R	PSW A	;row count	
0589	31	2 0550		JNZ	MOVROW	inext row	
0580	47		DRWZCU:		B, A	JZETO CUTSOF	
0580		, 3 OBDB		JMP	DCROW	; and delete cursor row	
	_				DIFFEREN		
0590		A SEFA	CRLR:		CROW	;calc crow to lrow	
0593		A 3FFC	EBCO.	LDA MOV	LRDW E, A		
0590	5F 93		FRCR	SUB	L	get the difference	
059		2 05A0		JNC	\$2	; within range	
0591		E 60	\$D2:	MVI		; over range	
0591				ADD	E	-	
0598				SUB	L		
059	B	7		ORA	Α	;clear carry	
0540			\$2:	RAR		;row diff /2	
05A	C,	9	DA05	RET			
			PAGE				
		ACRO-ASSEMBL	ER V2.0		PAGE	22	
CRTBO							
05A2				1011		5 0 D	
05A		A 3FA3	ADDCUR:		ress cur LINP	sor ;read leadin parameter	
054				MOV	A, H	·······	
0011							
05A		1 5010	CALCX:	LXI		h;calc x coordinate	
05A				CMP	D	00 (45) 0 (20	
054		A 05B3		JC	CX4FD	;00/4Fh=loc 0/79	
05A)				SUB CMP	D E	;50/7Fh offset to 0/2F	
05AI 05AI		B A 05B3		JC	CX4FD	;50/5Fh offset to 0/15	
				SUB	E	;60/7Fh offset to 0/31	
050	7.		CX4FD:	MOV	B, A	; then set cursor	
05B 05B	4						
05B 05B	4						
		E 1F	CALCY	MVI	A,01Fh		
05B 05B 05B	31 A	5	CALCY	ANA	L	; 0/1F, 20/3F, 40/5F, 60/7Fh	
05B 05B 05B 05B	3i Ai Fi	5 E 18	CALCY	ANA CPI	L O18h	;offset to 00/1Fh	
05B 05B 05B 05B 05B	3i A Fi Di	5 E 18 A 05BE	CALCY:	ANA CPI JC	L 018h CY17D	;offset to 00/1Fh ;00/17h=row 0/23	
058 058 058 058 058 058	3i A D D	5 E 18 A 05BE 6 18		ANA CPI JC SUI	L 018h CY17D 018h	;offset to 00/1Fh ;00/17h=row 0/23 ;18/1Fh=row 0/7	
05B 05B 05B 05B 05B	3i A D D	5 E 18 A 05BE	CALCY: CY17D:	ANA CPI JC	L 018h CY17D	;offset to 00/1Fh ;00/17h=row 0/23	
058 058 058 058 058 058 058	3i A D D D 2i	5 E 18 A 05BE 6 18 A 3FFE		ANA CPI JC SUI	L 018h CY17D 018h	; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0	
058 058 058 058 058 058	31 A D D D 21 0	5 E 18 A 05BE 6 18 A 3FFE 7	CY17D:	ANA CPI JC SUI LHLD	L 018h CY17D 018h	;offset to 00/1Fh ;00/17h=row 0/23 ;18/1Fh=row 0/7 ;offset first row	
05B 05B 05B 05B 05B 05B 05B 05B	3 A F D D D D D D C S S S S S S S S S S S S S	5 E 18 A 05BE 6 18 A 3FFE 7	CY17D:	ANA CPI JC SUI LHLD RLC	L 018h CY17D 018h FROW	; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2, msb=0 ; frow+offset	
058 058 058 058 058 058 058 058 058	3i A Di Di Di Di C S C S C S S C S S S S S S S S S S S	5 E 18 A 05BE 6 18 A 3FFE 7 5	CY17D:	ANA CPI JC SUI LHLD RLC ADD JC CPI	L O1Bh CY17D O1Bh FROW L ROSFFU 1ow RR4	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frow+offset ; >FFh 8+1</pre>	
058 058 058 058 058 058 058 058 058 058	3 A D D D D D C S C F	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB	CY17D: IRWOS:	ANA CPI JC SUI LHLD RLC ADD JC CPI JC	L 01Bh CY17D 01Bh FROW L ROSFFU 1ow RR4 ROSEOD	;offset to O0/1Fh ;O0/17h=row 0/23 ;I8/IFh=row 0/7 ;offset first row ;diff#2,msb=0 ;frow+offset ;>FFh B+1 ;less than EOh,ok	
058 058 058 058 058 058 058 058 058 058	3 A D D D D C C C C C C C C C C C C C C C	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB E E1	CY17D: IRWOS: ROSFFU:	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI	L 018h CY17D 018h FROW L ROSFFU 10w RR4 ROSEOD RWRG	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2, msb=0 ; frowtoffset ; >FFh 8+1 ; less than E0h, ok ; row range</pre>	
058 058 058 058 058 058 058 058 058 058	3 A D D D D D C C C C C C C C C C C C C C	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB E E1 A 05CD 6 60 2 3FFA	CY17D: IRWOS:	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI SUI STA	L 01Bh CY17D 01Bh FROW L ROSFFU 1ow RR4 ROSEOD	;offset to O0/1Fh ;O0/17h=row 0/23 ;I8/IFh=row 0/7 ;offset first row ;diff#2,msb=0 ;frow+offset ;>FFh B+1 ;less than EOh,ok	
058 058 058 058 058 058 058 058 058 056 056 056 056 056	3 A D D D D D C C C C C C C C C C C C C C	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB E E1 A 05CD 6 60 2 3FFA	CY17D: IRWOS: ROSFFU:	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI	L 018h CY17D 018h FROW L ROSFFU 10w RR4 ROSEOD RWRG	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2, msb=0 ; frowtoffset ; >FFh 8+1 ; less than E0h, ok ; row range</pre>	
058: 058: 058: 058: 058: 058: 058: 058:	3 A D D D D D C C C C C C C C C C C C C C	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB E E1 A 05CD 6 60 2 3FFA	CY17D: IRWOS: ROSFFU:	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI SUI STA	L 018h CY17D 018h FROW L ROSFFU 10w RR4 ROSEOD RWRG	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2, msb=0 ; frowtoffset ; >FFh 8+1 ; less than E0h, ok ; row range</pre>	
058 058 058 058 058 058 058 058 058 056 056 056 056 056 056 056	3 A D D D D D C C C C C C C C C C C C C C	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB E E1 A 05CD 6 60 2 3FFA	CY17D: IRWOS: ROSFFU: ROSEOD:	ANA CPI JC SUI LHLD RLC ADD JC CPI JC CPI JC SUI STA RET	L O1Bh CY17D O1Bh FROW ROSFFU 10w RR4 ROSEOD RWRG CROW	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frow+offset ; >FFh 8+1 ; less than EOh, ok ; row range ; then update crow</pre>	
058 058 058 058 058 058 058 058 058 056 056 056 056 056 056 056	C	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB E E1 A 05CD 6 60 2 3FFA	CY17D: IRWOS: ROSFFU: ROSEOD: ; {FUNC1	ANA CPI JC SUI LHLD RLC ADD JC CPI JC CPI JC SUI STA RET	L O1Bh CY17D O1Bh FROW L ROSFFU Iow RR4 ROSEOD RWRG CROW d cursor RDX	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2,msb=0 ; frowtoffset ; >FFh 8+1 ; less than EOh,ok ; row range ; then update crow ; read cursor x coord</pre>	
058: 058: 058: 058: 058: 058: 058: 058:	3 A D D D D D D D D C C	5 E 18 A 05BE 6 18 A 3FFE 7 5 5 A 05CB E E1 A 05CD 6 40 2 3FFA 9	CY17D: IRWOS: ROSFFU: ROSEOD: ; {FUNC1	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI STA RET	L O1Bh CY17D O1Bh FROW L ROSFFU Iow RR4 ROSEOD RWRG CROW d cursor RDX	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2, msb=0 ; frowtoffset ; >FFh 8+1 ; less than EOh, ok ; row range ; then update crow</pre>	
058: 058: 058: 058: 058: 058: 058: 056: 056: 056: 056: 056: 056: 056: 056	3 A D D D D D D D D C C	5 5 6 18 A 3FFE 7 7 7 5 A 05CB E E 1 A 05CD 6 4 0 5 2 3 7 7 7 7 7 7 7 7 7 7 7 7 7	CY17D: IRWOS: ROSFFU: ROSEOD: ; {FUNC1	ANA CPI JC SUI LHLD ADD JC CPI JC SUI STA RET CALL CALL	L O1Bh CY17D O1Bh FROW L ROSFFU Iow RR4 ROSEOD RWRG CROW d cursor RDX WTACEA	<pre>; offset to O0/1Fh ; O0/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frow+offset ; >FFh B+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace</pre>	
058: 058: 058: 058: 058: 058: 058: 058:	A A F D D D D D D D C C C C C C C	5 6 18 A 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; {FUNC1	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI SUI STA RET TION} rea CALL CALL	L O1Bh CY17D O1Bh FRDW L ROSFFUU 10w RR4 ROSEOD RWRG CROW d cursor RDX WTACEA RDY	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2,msb=0 ; frowtoffset ; >FFh 8+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace ; read cursor y coord</pre>	
058: 058: 058: 058: 058: 058: 058: 050: 050	A A F D D D D D D D C C C C C C C	5 E 18 A 05BE 6 18 A 3FFE 7 5 A 05CB E E1 A 05CD 6 60 2 3FFA 9 D 05E0 D 006D	CY17D: IRWOS: ROSFFU: ROSEOD: ; {FUNC1	ANA CPI JC SUI LHLD ADD JC CPI JC SUI SUI STA RET CALL CALL	L 01Bh CY17D 01Bh FRDW L ROSFFUU 10w RR4 ROSEOD RWRG CROW d cursor RDX WTACEA RDY	<pre>; offset to O0/1Fh ; O0/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frow+offset ; >FFh B+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace</pre>	
058: 058: 058: 058: 058: 058: 058: 056: 056: 056: 056: 056: 056: 056: 056	SI FDD DD DD DD CC CC CC	5 6 18 A 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; {FUNC1	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI SUI STA RET TION} rea CALL CALL	L 01Bh CY17D 01Bh FRDW L ROSFFUU 10w RR4 ROSEOD RWRG CROW d cursor RDX WTACEA RDY	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2,msb=0 ; frowtoffset ; >FFh 8+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace ; read cursor y coord</pre>	
05B: 05B: 05B: 05B: 05B: 05B: 05C: 05C: 05C: 05C: 05C: 05C: 05C: 05C		5 6 18 4 05BE 6 18 4 05FE 7 5 5 5 6 05CB 6 4 05CD 6 4 05CD 6 4 05CD 6 4 05CD 6 4 05CD 6 18 7 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; (FUNCT RDCUR:	ANA CPI JC SUI LHLD ADD JC CPI JC SUI STA RET CALL CALL CALL CALL CALL	L OIBh CY17D OIBh FROW L ROSFFU Iow RR4 ROSFO RWRG CROW d CUTSOT RDX WTACEA CRACE	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frow+offset ; >FFh B+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; urite to ace ; read cursor y coord ; write to ace</pre>	
058: 058- 058- 058- 058- 058- 058- 058- 058-	FFDD22 0BBFDD3C CCCC7	5 6 18 A 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; {FUNC1	ANA CPI JC SUI LHLD RLC ADD JC CPI JC SUI STA RET TION} rea CALL CALL CALL CALL CALL CALL CALL	L O1Bh CY17D O1Bh FRDW L ROSFU 10w RR4 ROSEOD RWRG CROW MTACEA RDY WTACEA CRACE A. B	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frow+offset ;)>FFh 8+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace ; read cursor y coord ; write to ace ; cr for termination</pre>	
058: 058: 058: 058: 058: 058: 058: 058:	3 A FF D D D D D D D D D D D D D D D D D	5 E 18 A O5BE 6 18 A O5FE 7 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; (FUNCT RDCUR:	ANA CPI JC SUI LHLD ADD JC CPI JC SUI STA RET CALL CALL CALL CALL CALL	L OIBh CY17D OIBh FROW L ROSFFU Iow RR4 ROSFO RWRG CROW d CUTSOT RDX WTACEA CRACE	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frow+offset ; >FFh B+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; urite to ace ; read cursor y coord ; write to ace</pre>	
058 058 058 058 058 058 058 058 058 058	3A FFIDD 22 0B B D D 22 0 B C C C C C C C C C C F D D D 22 0 B C C C C C C C C C C C C C C C C C C	5 5 6 18 A 3FFE 7 7 7 5 A 05CB E E 1 A 05CD 6 4 0 5 2 3 0 5 5 2 3 0 5 5 2 3 5 5 2 3 5 5 2 3 5 5 2 3 5 5 2 3 5 5 2 3 5 5 2 3 5 5 2 3 5 5 2 3 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; (FUNCT RDCUR:	ANA CPI JC SUI LHLD ADD JC CPI JC SUI SUI SUI SUI SUI SUI CALL CALL CALL CALL CALL CALL CALL CAL	L O1Bh CY17D O1Bh FRDW L ROSFU 10w RR4 ROSEOD RWRG CROW MTACEA RDY WTACEA CRACE A. B	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff#2, msb=0 ; frowtoffset ; >FFh 8+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace ; read cursor y coord ; write to ace ; cor for termination ; if 0/1Fh add offset</pre>	
058: 058: 058: 058: 058: 058: 058: 058:	3 A A FI D D 2 2 0 8 F D D 2 2 0 8 F C C C C C C C C C C C C C C C C C C	5 E 18 A O5BE 6 18 A O5FE 7 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; (FUNCT RDCUR:	ANA CPI JC SUI LHLD RLC ADD JC CPI SUI STA RET CALL CALL CALL CALL CALL CALL CALL CAL	L O1Bh CY17D O1Bh FRDW L ROSFFU 10w RR4 ROSEOD RWR0 CROW WTACEA RDY WTACEA CRACE A.B O2Oh	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frowtoffset ; >>FFh B+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace ; read cursor y coord ; write to ace ; read cursor y coord ; write to ace ; cr for termination ; if 0/1Fh add offset ; 20/4Fh=cursor 1oc 32/79</pre>	
058: 058: 058: 058: 058: 058: 058: 058:	3 A A FI D D 2 2 0 8 F D D 2 2 0 8 F C C C C C C C C C C C C C C C C C C	5 6 18 4 05BE 6 18 4 18 5 5 5 5 5 5 5 5 5 5 5 5 5	CY17D: IRWOS: ROSFFU: ROSEOD: ; (FUNCT RDCUR:	ANA CPI JC SUI LHLD RLC ADD JC CPI SUI SUI SUI STA RET ION} rea CALL CALL CALL CALL CALL CALL CALL CAL	L O1Bh CY17D O1Bh FRDW L ROSFFU 10w RR4 ROSEOD RWR0 CROW WTACEA RDY WTACEA CRACE A.B O2Oh	<pre>; offset to 00/1Fh ; 00/17h=row 0/23 ; 18/1Fh=row 0/7 ; offset first row ; diff*2,msb=0 ; frowtoffset ; >>FFh B+1 ; less than EOh, ok ; row range ; then update crow ; read cursor x coord ; write to ace ; read cursor y coord ; write to ace ; read cursor y coord ; write to ace ; cr for termination ; if 0/1Fh add offset ; 20/4Fh=cursor 1oc 32/79</pre>	TL/F/5869-

$\begin{array}{cccccccccccccccccccccccccccccccccccc$			
OSEA 3.3 3FFA LDA CROW OSED CD 0596 CALL FRCR ;calc frow/crow diff OSFO C6 60 ADI 060h ;60/77h=row 0/23 OSFO C6 60 ADI 060h ;60/77h=row 0/23 OSF2 C9 RET PAGE 23 CRTB01 05F3 i6 NPAGE: MVI D.030h ;page offset OSFS 21 3FFA LX1 H.CROW ;page offset 0575 OSF8 F3 DI DI 051h ;GFINCTION> next page offset OSF6 F2 NPLP: MOV A.M ;GSF9 OSIH ;GSF9 OSF6 F2 SUB D ;a<=B1, B2=B0h			
05EA 3A 3FFA LDA CRUW 03ED CD 0596 CALL FRCM ;calc frow/crow diff 05F0 C6 60 ADI 060h ;60/77h=row 0/23 05F2 C9 RET PAGE 23 FARPLEX STARPLEX MACRO-ASSEMBLER V2.0 PAGE 23 OSF3 16 30 NPAGE: MVI D.030h ; page offset OSF3 16 30 NPAGE: MVI D.030h ; page offset OSF3 16 30 NPAGE: MVI D.030h ; page offset OSF3 16 30 NPAGE: MVI D.030h ; page offset OSF6 21 3FFA LX1 H.CRUM OSF6 72 SUB D ; a<< B1.82-B0h			
$\begin{array}{c c c c c c c c c c c c c c c c c c c $			
05F0 C6 60 ADI 060h ;60/77h=row 0/23 PAGE STARPLEX MACRO-ASSEMBLER V2.0 PAGE 23 CTUDE next page (FUNCTION> next page 05F3 16 30 NPAGE: MVI b,030h ; page offset 05F3 16 30 NPAGE: MVI b,030h ; page offset 05F3 16 30 NPAGE: MVI b,030h ; page offset 05F5 21 3FFA LX1 H,CROW OSF5 21 3FFA LX1 H,CROW OSF6 72 NPLF MOV A,M OSF7 FB DI ; a>=B1,B2==B0h OSF6 92 SUB D ; a>=B1,B2==E0h OS60 92 SUB D ; a>=B1,B2==E0h OS67 JM NPLT ; a>=B1,B2==E0h OS60 92 SUB D ; a>=B1,B2==E0h OS60			
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $			
PAGE STARPLEX MACRD-ASSEMBLER V2.0 PAQE 23 CRT901 OSF3 (FUNCTION) next page STARPLEX OSF3 (FUNCTION) next page OSF3 16 30 OSF3 16 30 OSF3 (FUNCTION) next page OSF3 16 30 OSF3 10 OSF3 10 OSF3 10 OSF3 13FA LX1 H, CRDW OSF3 PTE NPLF: MOV A, M OSF6 FE DI OSF6 P2 SUB D asset colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspa= Colspan="2" Colspan="2"Colspan="2"Col			
$\begin{array}{c c c c c c c c c c c c c c c c c c c $			
CRT801 05F3 ; {FUNCTION} next page 05F3 16 30 NPAGE: MVI D,030h ; page offset 05F3 21 3FFA LXI H, CROW 05F5 21 3FFA DI 05F7 7E NPLP: MOV A,M 05F6 DA JC NPLT ; a< B1,82-B0h			
CRT801 05F3 ; {FUNCTION} next page 05F3 16 30 NPAGE: MVI D,030h ; page offset 05F3 21 3FFA LXI H, CROW 05F5 21 3FFA DI 05F7 7E NPLP: MOV A,M 05F6 DA JC NPLT ; a< B1,82-B0h			
OSF3 ifFUNCTION> next page OSF3 i6 30 NPAGE: MVI D.030h ;page offset OSF5 21 3FFA LXI H.CROW OSF5 21 3FFA LXI H.CROW OSF5 21 3FFA LXI H.CROW OSF7 PE NPLP: MOV A,M OSF7 PE NPLF: MOV A,M OSF7 PE NPLF: MOV A,M OSF7 PE NPLF: MOV A,M OSF7 PE NPLT: ADD OS60 PE SUB D ; do crow/lrow/frow OS60 PE OS60 PE IMOV M.A OS60 PE OS60 PE PE OS60 PE IMOV M.A OS60 PE <td <="" colspan="2" td=""><td></td></td>	<td></td>		
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $			
05F3 16 30 NPAGE: MVI D, 030h ;page offset 05F5 21 3FFA LX1 H, CROW			
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$			
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$			
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$			
05FC DA 0601 JC NPLT ;a < B1, 82-B0h			
05FF 92 SUB D ;a>=B1,B2-E0h 0600 92 SUB D 0601 B2 NPLT: ADD D 0602 77 MOV M.A 0603 2C INR L 0604 2C INR L 0605 FA 05F7 JM NPLP 0608 FB EI 0609 C7 RET 0600 E5 PUSH H 0611 13 TDADSP: INX D 0611 13 TDADSP: INX D ;next character 0612 2D DCR L ;end of row? G 0613 CC 067A C2 DCREL ;down one row, get 1st loc 0616 14			
0600 92 5UB D 0601 B2 NPLT: ADD D 0602 77 MOV M, A 0603 2C INR L 0604 2C INR L 0605 FA 05F9 JM NPLP 0608 FB EI 0609 C9 RET // 0600 E5 PUSH 0601 13 TDADSP: 0611 13 TDADSP: 0612 2D DCR 0613 CC 067A C2 0616 1A LDAX D			
0601 B2 NPLT: ADD D 0602 77 MOV M, A 0603 2C INR L ;do crow/lrow/frow 0603 2C INR L ;do crow/lrow/frow 0604 2C INR L ;do crow/lrow/frow 0605 FA 05F9 JM NPLP ;if pass frow, end 0608 FB EI 0609 C9 RET 0604 2A JFFA TAB: LHLD CRUW 0606 CD 0667 CALL SCATT 0606 CD 0667 CALL SCATT 0611 13 TDADSP: INX D ; next character 0612 2D DCR L ; end of row? 0613 0613 CC 0613 CC 067A C2 DRCFL ; down one row, get 1st loc 0616 1A LDAX D ; read character			
0602 77 MOV M, A 0603 2C INR L ;do crow/lrow/frow 0604 2C INR L .do crow/lrow/frow 0604 2C INR L .do crow/lrow/frow 0605 FA 05F9 JM NPLP ;if pass frow, end 0608 FB EI			
0603 2C INR L ; do crow/lrow/frow 0604 2C INR L 0605 FA 05F9 JM NPLP ; if pass frow, end 0608 FB EI			
0604 2C INR L 0605 FA 05F9 JM NPLP ; if pass frow, end 0608 FB EI 0609 C9 RET 0604 2A 3FFA TAB: LHLD 0604 2A 3FFA TAB: LHLD 0604 C0 E5 PUSH 0606 CD 0667 CALL 0611 13 TDADSP: INX D 0612 2D DCR L ;end of row? 0613 CC 067A C2 DRCFL ;down one row, get 1st loc 0616 1A LDAX D ;read character			
0605 FA 05F9 JM NPLP ; if pass frow, end 0608 FB EI 0609 C9 RET , (FUNCTION) tab , 0604 2A 3FFA TAB: 0605 C9 RET 0606 C3 3FFA TAB: 0606 C5 PUSH 0606 C5 PUSH 0606 C4LL SCATT 0611 13 TDADSP: INX 0612 2D DCR L 0613 CC 067A C2 DRCFL ; down one row, get 1st loc 0616 1A LDAX D ; read character			
0608 FB EI 0609 C9 RET ifFUNCTION> tab iffunctions 060A 2A 3FFA 060D ES PUSH 060E CD 0667 060E CD 0667 0611 13 TDADSP: 0612 2D DCR 0613 CC 067A 0616 1A LDAX D ; read character 0616 1A LDAX			
0609 C9 RET ;{FUNCTION} tab ; 060A 2A 3FFA TAB: LHLD CRUW 060D E5 PUSH H ; save crow 060E CD 0667 CALL SCATT 0611 13 TDADSP: INX D ; next character 0612 2D DCR L ; end of row? 0613 CC 067A C2 DRCFL ; down one row, get 1st loc 0616 1A LDAX D ; read character			
;{FUNCTION} tab O60A 2A 3FFA TAB: LHLD CRDW O60D E5 PUSH H ;save crow O60E CD 0667 CALL SCATT O611 13 TDADSP: INX D ;next character O612 2D DCR L ;end of row? O613 CC 067A CZ DRCFL ;down one row, get 1st loc O616 1A LDAX D ;read character			
060A 2A 3FFA TAB: LHLD CRDW 060D E5 PUSH H ; save crow 060E CD 0667 CALL SCATT 0611 13 TDADSP: INX D ; next character 0612 2D DCR L ; end of row? 0613 CC 067A CZ DRCFL ; down one row, get 1st loc 0616 1A LDAX D ; read character			
060A 2A 3FFA TAB: LHLD CRDW 060D E5 PUSH H ; save crow 060E CD 0667 CALL SCATT 0611 13 TDADSP: INX D ; next character 0612 2D DCR L ; end of row? 0613 CC 067A CZ DRCFL ; down one row, get 1st loc 0616 1A LDAX D ; read character			
060D E5 PUSH H ; save crow 060E CD 0667 CALL SCATT 0611 13 TDADSP: INX D ; next character 0612 2D DCR L ; end of row? 0613 CC 067A CZ DRCFL ; down one row, get 1st loc 0616 1A LDAX D ; read character			
060E CD 0667 CALL SCATT 0611 13 TDADSP: INX D ; next character 0612 2D DCR L ; end of row? 0613 CC 067A CZ DRCFL ; down one row, get 1st loc 0616 1A LDAX D ; read character			
0611 13 TDADSP: INX D ;next character 0612 2D DCR L ;end of row? 0613 CC 067A CZ DRCFL ;down one row, get ist loc 0616 1A LDAX D ;read character			
O612 2D DCR L ;end of row? O613 CC O67A CZ DRCFL ;down one row, get 1st loc O616 1A LDAX D ;read character	-		
O612 2D DCR L ;end of row? O613 CC O67A CZ DRCFL ;down one row, get 1st loc O616 1A LDAX D ;read character			
0616 1A LDAX D ;read character			
0616 1A LDAX D ; read character			
0617 84 ADD H ;check attribute			
0618 FA 0635 JM TSATT ;diff, find same attrib			
061B FE 20 CPI SPC ; space?			
061D C2 0611 JNZ TDAOSP ; loop until space			
0620 13 TSANSP: INX D ;next character			
0621 2D DCR L ; end of row?			
0622 CC 067A CZ DRCFL ; down one row, get 1st loc	1		
0625 1A LDAX D ; read character			
0626 84 ADD H ;check attribute			
0627 FA 0635 JM TSATT / diff, find same attrib			
062A FE 20 CPI SPC ; non space? 062C CA 0620 JZ TSANSP ; loop until non space			
062C CA 0620 JZ TSANSP ;loop until non space 062F D1 TMCUR: PDP D ;remove saved crow			
0632 95 SUB L ; calc cursor location			
0634 C7 RET			
0635 13 TSATT: INX D ; next character			
0636 2D DCR L ; end of row?			
0637 CC 067A CZ DRCFL ; down one row, get 1st loc			
063A 1A LDAX D ;read character			
O63B 84 ADD H ;check attribute			
063C FA 0635 JM TSATT ;loop until same attrib			
063F C3 062F JMP TMCUR ; maye cursor			
PAGE			
STARPLEX MACRO-ASSEMBLER V2.0 PAGE 24			
CRT801			
0642			
;{FUNCTION} clear fore/background to space	1		
0642 2A 3FFA CFB: LHLD CROW			
0645 24 INR H			
0646 E5 PUSH H ; save			
0647 CD 0667 CALL SCATT			
064A 1A CFBLP: LDAX D ; read character			
064B 84 ADD H ; test attribute			
064C FA 0653 JM CFBDIF ;diff attrib 064F 3E 20 MVI A/SPC ;if same attrib;			
0651 B4 DRA H ; put a space 0652 12 STAX D			
0653 13 CFBDIF: INX D ;next character	1		
0654 2D DCR L ;character counter~1			
0655 C2 064A JNZ CFBLP ; until finish 80 char			
0658 E1 POP H ; saved crow			
0659 3A 3FFC LDA LROW			
065C BD CMP L ;row=lrow?			
065D CB RZ ; no more			
065E 6E MOV L.M ;else next row			
065F E5 PUSH H ; save			
0660 CD 0681 CALL \$3			
0663 C3 064A JMP CFBLP			
0666 C9 RET			
TL/F/5869-21			

0.417			CHAR CO			
0667	CD 0174	SCATT:	CALL XCHG		get cursor location	
066A 066B	EB 21 8050		LXI		;put in de s;mask/char count	
066E	3A 3F20		LDA	FBG		
0671	A4		ANA	н		
0672	67		MOV	н, а		
0673	22 3FA5		SHLD		;save count,attrib	
0676	7D		MOV	A, L		
0677	90		SUB MOV		get cur to end diff	
0678 0679	6F C9		RET	L, A	put in l	
00//	0,					
				ND GET AD	DR ON 1ST COLUMN	
067A	CD 0463	DRCFL:			idown cursor one row	
067D	CA 0033		JZ		;crow=lrow, no tab	
0680	6E		MOV		inext row	
0681	25	\$3:	DCR	н		
0682	2B	\$1:	DCX	н		
0683	56		MOV		;addr high	
0684	28		DCX	H		
0685	5E		MOV		addr low	
0686 0689	2A 3FA5 C9		LHLD RET	LINP+2	;count and attrib	
0007	67	PAGE	RE!			
STARPI EY	MACRD-ASSEMBL			PAGE	25	
CRT801	MONO HOOCHDL	LIV TE. U		, HVL		
068A						
		; (FUNCT	ION} gra	phics		
068A	3A 3FA3	GRAPH:	LDA	LINP	read leadin parameter	
068D	21 107E	LGPARA:		H, 0107Eh	1	
0690	3D		DCR	A		
0691	E6 3F		ANI	03Fh		
0693	BC DA D(AD		CMP		;h=010h ;"A" to "P"	
0694	DA 06AD		JC CPI			
0697 0699	FE 3E CA 0682		JZ	ogen ≸G3	; delete	
0690	94		SUB		; h=010h	
0690	FE OF		CPI		; space?	
069F	CA 0682		JZ		space	
06A2	BC		CMP	н	; h=010h	
06A3	DA 06AC		JC	\$G2	i> "Q"	
0646	FE 20		CPI	020h		
0648	DA O6AD		JC		;"a" to "p"	
06AB	20		INR		;set 1 to 7Fh	
06AC	7D	\$G2:	MOV		L=7Eh or 7Fh	
06AD	CD 06D0	\$G1:	CALL ADI	STSCN 01Eh	;put symbol to screen	
0680	C6 1E	\$G3:	SUI	01Eh		
06B2 06B4	D6 1E 32 3FF3	*90.	STA		;for non/destructive move	
					, Hon, descructive move	
0687	3E 80	ENGRPH:		A, 080h		
0689	C3 06BD		JMP	DEGRPH+1		
			ION} def		nics	
O6BC	AF	DEGRPH:		A		
06BD	32 3FEF		STA	GECNTL		
0400	C9		RET			
		CHECK	GRAPHICS	MODE AND	PUT SYMBOL TO SCREEN	
0601	3A 3FEF	CHKGM:	LDA	GECNTL		
0604	B7		ORA		graphics mode?	
0605	CB		RZ		ino, rtn to func	
0606	3A 3FF3		LDA	GSYMBL		
0609	B7		ORA		inon destructive move?	
06CA	FB		RM		; yes	
06CB	21 3F20	STSP:	LXI	H, FBG		
06CE	AE		XRA		;get attribute	
2702	AD	OTOON	XRA		;remove space code	
06CF	CD 0174	STSCN:	CALL MOV	CURLOC	units to score	
0600			RET	19.4	;write to screen	
06D0 06D3	77		N G 1			
0600		PAGE				
06D0 06D3 06D4	77 C9			PACE	26	
06D0 06D3 06D4 STARPLEX	77			PAGE	26	
06D0 06D3 06D4 STARPLEX CRT801	77 C9			PAGE	26	
06D0 06D3 06D4 STARPLEX	77 C9	ER V2.0			26 REEN (LOCAL)	
06D0 06D3 06D4 STARPLEX CRT801	77 C9	ER V2.0	RAPHIC ME			
06D0 06D3 06D4 STARPLEX CRT801 06D5	77 C9 MACRO-ASSEMBL CD 070C	ER V2.0		NU TO SCR RUADD	REEN (LOCAL)	
06D0 06D3 06D4 STARPLEX CRT801 06D5 06D5	77 C9 MACRO-ASSEMBL	ER V2.0	CALL	NU TO SCR RUADD P38SYM		

06DE	CD 070C		CALL	RUADD		
06E1	CD 0700		CALL	P325YM	print ascii Oh/Fh	
06E4	F6 7E		ORI	07Eh	;print ascii 7Eh	
06E6	CD 06F7		CALL	P65YM	;4 spaces	
06E9	EE 30		XRI	030h	;print ascii 10/1Fh	
06EB	CD 0700		CALL	P32SYM		
06EE 06F0	F6 7F C3 06F7		ORI JMP	07Fh P6SYM	;print ascii 7Fh ;4 spaces	
0010			011	100111) - spaces	
06F3	F6 41	P38SYM:	ORI	041h	;change symbol	
06F5	16 10		MVI	D, 16	; counter	
06F7	14	P6SYM:	INR	D		
06F8	CD 0702		CALL	PGMLP		
06FB	16 04	P4SP:	MVI	D, 4	L .	
06FD	C3 04C4		JMP	DRLLP-3	;put 4 spaces	
0700	16 10	P32SYM:	MVI	D, 16	;print 32 symbols	
0702	77	PGMLP:	MOV	M, A		
0703	23		INX	н		
0704	73		MOV	M, E	;FBG	
0705	23		INX	н		
0706	30		INR	Α	;next symbol	
0707	15		DCR	D		
0708	C2 0702		JNZ	PGMLP		
070B	C9		RET			
0700	CD 0472	RUADD:	CALL	ROLUP	jup one row	
070F	26 00		MVI	H, 0	, op one , op	
0711	6F		MOV	L, A	;hl≏lrow	
0712	CD 0682		CALL	\$1	;lrow 1st loc	
0715	EB		XCHG			
0716	CD 06FB		CALL	P4SP	; put 4 spaces	
0719	5F		MOV	E, A	FBG	
071A 071C	E6 80 C9		ANI RET	080h	;get attribute	
0/10	07	PAGE				
STARPLI	EX MACRO-ASSEM	BLER V2. 0		PAGE	27	
CRTB01						
071D						
0/10						
				ansmit a :		
071D	06 00	; {FUNCT SNDLNE:	MVI	в, о	;set cursor to 1st loc	
071D 071F	CD 0174		MVI Call	B, O CURLOC	;set cursor to 1st loc ;get row start	
071D 071F 0722	CD 0174 E5		MVI CALL PUSH	B,0 CURLOC H	;set cursor to 1st loc ;get row start ;save row start	
071D 071F 0722 0723	CD 0174 E5 11 004F		MVI CALL PUSH LXI	B, 0 CURLOC H D, 79	;set cursor to 1st loc ;get row start ;save row start ;find row end	
071D 071F 0722 0723 0726	CD 0174 E5 11 004F 19		MVI CALL PUSH LXI DAD	B, O CURLOC H D, 79 D	;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry	
071D 071F 0722 0723 0726 0727	CD 0174 E5 11 004F 19 16 40	SNDLNE:	MVI CALL PUSH LXI DAD MVI	B,0 CURLOC H D,79 D D,040h	;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry ;rotated space	
071D 071F 0722 0723 0726	CD 0174 E5 11 004F 19		MVI CALL PUSH LXI DAD MVI	B, O CURLOC H D, 79 D	;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry ;rotated space ;read char	
071D 071F 0722 0723 0726 0727 0727	CD 0174 E5 11 004F 19 16 40 7E	SNDLNE:	MVI CALL PUSH LXI DAD MVI MOV	B,0 CURLOC H D,79 D D,040h	;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry ;rotated space	
071D 071F 0722 0723 0726 0727 0729 0729 0729	CD 0174 E5 11 004F 19 16 40 7E 17	SNDLNE:	MVI CALL PUSH LXI DAD MVI MOV RAL	B, 0 CURLOC H D, 79 D D, 040h A, M	;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry ;rotated space ;read char ;mask out msb	
071D 071F 0722 0723 0726 0727 0729 072A 072B 072C 072B 072C 072F	CD 0174 E5 11 004F 17 16 40 7E 17 AA	SNDLNE:	MVI CALL PUSH LXI DAD MVI RAL XRA JNZ DCX	B, O CURLOC H D, 79 D D, 040h A, M D LNSP H	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;h1=row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0726 0726 0727	CD 0174 E5 11 004F 17 16 40 7E 17 AA C2 0734 28 1D	SNDLNE:	MVI CALL PUSH LXI DAD MVI RAL XRA JNZ DCX DCR	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E	; set cursor to 1st loc ; get row start ; save row start ; find row end ; h1=row end, clear carry ; rotated space ; read char ; mask out msb ; space; , clear carry ; until find a non-space	
071D 071F 0722 0723 0726 0727 0729 0728 0728 0728 0726 0726 0731	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 1D C2 0729	SNDLNE:	MVI CALL PUSH LXI DAD MVI MOV RAL XRA JNZ DCX DCR JNZ	B, O CURLOC H D, 79 D, O40h A, M D LNSP H E LENDLP	<pre>;set cursor to ist loc ;get row start ;find row end ;hl≖row end ;rotated space ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop</pre>	
071D 071F 0722 0723 0726 0727 0729 072A 072B 072C 072F 0730 0731	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 1D C2 0729 E1	SNDLNE: LENDLP: LNSP:	MVI CALL PUSH LXI DAD MVI MOV RAL XRA JNZ DCX DCX DCR JNZ POP	B, 0 CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl≖row start again</pre>	
071D 071F 0722 0723 0726 0727 0729 0728 0728 0726 0726 0726 0730 0731 0731 0734 0735	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 1D C2 0729 E1 56	SNDLNE:	MVI CALL PUSH LXI DAD MVI MOV RAL XRA JNZ DCX DCR JNZ POP MDV	B, 0 CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space; clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read charatter</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0727 0729 0726 0730 0731 0734 0735 0734	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 1D C2 0729 E1 56 3E 60	SNDLNE: LENDLP: LNSP:	MVI CALL PUSH LXI DAD MVI MUV RAL XRA JNZ DCX DCR JNZ DCR JNZ MUV MVI	B, 0 CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl≖row start again</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0726 0726 0730 0731 0731 0734 0735 0736 0735	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 10 C2 0729 E1 56 3E 60 A2	SNDLNE: LENDLP: LNSP:	MVI CALL PUSH LXI DAD MVI MOV RAL XRA JNZ DCX DCX DCX DCX DCX DCX DCX ANA	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl≖row end, clear carry ;rotated space ;read char ;mask out msb ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes</pre>	
071D 071F 0722 0723 0726 0727 0729 0728 0728 0728 0726 0730 0731 0734 0735 0736 0736 0736 0736	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B C2 0734 2B C2 0729 E1 56 56 3E 60 A2 C2 073E	SNDLNE: LENDLP: LNSP:	MVI CALL PUSH DAD MVI RAL JNZ JNZ DCX DCX DCX DCX DCX DCX DCX DCX JNZ	B, 0 CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H G N M A, 060h D SNCNTL	<pre>;set cursor to ist loc ;get row start ;find row end ;hl≖row end ;hl≖row end, clear carry ;rotated space ;read char ;mask out msb ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0728 0726 0727 0730 0731 0734 0735 0734 0735 0736 0738 0739 0736	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 28 10 C2 0729 E1 56 3E 60 A2 C2 073E 16 2A	SNDLNE: LENDLP: LNSP: SNDLP:	MVIL CALL PUSH LXI DAD MVI XRA XRA JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*"	<pre>;set cursor to ist loc ;get row start ;save row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code,send "*" instead</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0726 0727 0730 0731 0734 0735 0736 0735 0736 0739 0736	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 10 C2 0734 E1 56 3E 60 A2 C2 073E 16 2A CD 006E	SNDLNE: LENDLP: LNSP:	MVI CALL PUSH LXI DAD MVI RAL JNZ XRA JNZ DCR JNZ DCR JNZ DCR JNZ MVI ANA JNZ GALL	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*"	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code,send "*" instead ;write to ace</pre>	
071D 071F 0722 0723 0726 0727 0729 0728 0728 0728 0726 0731 0734 0735 0736 0736 0738 0736 0738 0736 0738 0736 0738	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 10 C2 0729 E1 56 3E 60 A2 C2 073E 16 2A CD 006E 7E	SNDLNE: LENDLP: LNSP: SNDLP:	MVI CALL PUSH LXI DAD MVI RAL JNZ JNZ JNZ JNZ JNZ JNZ JNZ JNZ JNZ JNZ	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*"	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;untl code,send "*" instead ;wrete to ace ;read char again</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0726 0727 0730 0731 0734 0735 0736 0735 0736 0739 0736	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 10 C2 0734 E1 56 3E 60 A2 C2 073E 16 2A CD 006E	SNDLNE: LENDLP: LNSP: SNDLP:	MVI CALL PUSH LXI DAD MVI RAL JNZ XRA JNZ DCR JNZ DCR JNZ DCR JNZ MVI ANA JNZ GALL	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code,send "*" instead ;write to ace</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0726 0726 0730 0731 0734 0735 0736 0735 0736 0737 0738 0739 0739	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 10 C2 0729 E1 56 3E 60 A2 C2 073E 16 2A CD 006E 7E EE 80	SNDLNE: LENDLP: LNSP: SNDLP:	MVI CALL PUSH LXI DAD MVI MAVI MAVI XRA JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX XRI ANA JNZ ANA ZNZ XRI	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M OBOh	<pre>;set cursor to ist loc ;get row start ;save row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0727 0730 0731 0734 0735 0736 0735 0736 0739 0739 0739 0739 0736 0734 0744 0744	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 1D C2 0729 E1 56 3E 60 A2 C2 073E 16 2A C2 073E 16 2A C2 006E 7E EE 80 77	SNDLNE: LENDLP: LNSP: SNDLP:	MVI CALH LXI DAD MVI MOVI RAL JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 77 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M OBOh M, A	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code,send "*" instead ;write to ace ;read char again ;invert attribute ;store back</pre>	
071D 071F 0722 0723 0724 0727 0729 0724 0728 0726 0726 0736 0731 0734 0735 0734 0735 0734 0735 0736 0738 0736 0732 0732 0732 0732 0732 0732 0734	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 1D C2 0729 E1 56 3E 60 A2 C2 073E 16 2A CD 006E 7E E 80 77 72	SNDLNE: LENDLP: LNSP: SNDLP:	MVI CALH LXI DAD MVI MAU XRA JNZ DCX DCZ DCZ DCZ DCZ DCZ DCZ VDV ANA ZNI CALL XRI XRI XRI XRI XRI XRI XRI XRI XRI XRI	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D, "*" WTACED A, M O80h M, A	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code,send "*" instead ;write to ace ;read char again ;invert attribute ;store back</pre>	
071D 071F 0722 0723 0724 0727 0729 0728 0728 0726 0730 0731 0731 0734 0735 0736 0736 0737 0736 0738 0739 0736 0738 0734 0741 0742 0744	CD 0174 E5 11 004F 17 7E 17 AA C2 0734 2B 10 C2 0729 E1 56 3E 60 A2 C2 073E 16 2A C2 073E 16 2A C2 073E 16 2A C2 006E 7E EE 80 77 72 23 20 735 CD 006B	SNDLNE: LENDLP: LNSP: SNDLP:	MVI CALH LXI DAD MVI MOVI MVI MVI JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M OBOh M, A H E SNDLP SNDCR	<pre>;set cursor to 1st loc ;get row start ;save row start ;find row end ;h1=row end, clear carry ;rotated space ;read char ;mask out msb ;space?, clear carry ;until find a non-space ;repeat loop ;h1=row start again ;read character ;screen all cnt1 codes ;not control code ;ent1 code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character</pre>	
071D 0722 0723 0726 0727 0729 0724 0728 0726 0727 0730 0731 0734 0735 0736 0736 0738 0736 0738 0736 0738 0736 0738 0736 0738 0736 0738 0736 0741 0742 0744 0745 0744	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 28 C2 0729 E1 56 3E 60 A2 C2 0729 E1 56 3E 60 A2 C2 073E 16 2A CD 006E 77 23 10 F2 0735 CD 006B 3A 3FF1	SNDLNE: LENDLP: SNDLP: SNCNTL:	MVI CALL PUSH LXI DAD MVI MAL XRA JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D, "*" WTACED A, M OBOh M, A H E SNDLP SNDCR LOCLM	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, ;near char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl=row start again ;reped character ;screen all cntl codes ;not control code ;cntl code,send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0726 0730 0731 0734 0735 0736 0736 0735 0736 0735 0736 0735 0736 0737 0736 0737 0744 0745 0744 0745 0744 0745	CD 0174 E5 11 004F 17 A A C2 0734 28 10 C2 0734 28 C2 0734 28 C2 0735 C2 0735 CD 004B 3A 3FF1 B7	SNDLNE: LENDLP: SNDLP: SNCNTL:	MVI CALL PUSH LXI DAD MVI MOV XRA JNZ DCX DCX DCX DCX DCX DCX ZNZ DCX DCX ZNZ ZNZ ZNZ ZNZ ZNZ ZNZ ZNZ ZNZ ZNZ ZN	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M OBOh M, A H E SNDLP SNDCR	<pre>;set cursor to ist loc ;get row start ;save row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination ;local?</pre>	
071D 0722 0723 0726 0727 0729 0724 0728 0726 0727 0729 0730 0731 0734 0735 0736 0738 0736 0738 0736 0738 0736 0738 0736 0738 0736 0738 0736 0741 0742 0744 0745 0744	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 28 C2 0729 E1 56 3E 60 A2 C2 0729 E1 56 3E 60 A2 C2 073E 16 2A CD 006E 77 23 10 F2 0735 CD 006B 3A 3FF1	SNDLNE: LENDLP: SNDLP: SNCNTL:	MVI CALL PUSH LXI DAD MVI MAL XRA JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D, "*" WTACED A, M OBOh M, A H E SNDLP SNDCR LOCLM	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, ;near char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl=row start again ;reped character ;screen all cntl codes ;not control code ;cntl code,send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination</pre>	
071D 071F 0722 0723 0726 0727 0729 0720 0728 0726 0726 0726 0726 0727 0730 0736 0736 0736 0736 0736 0737 0736 0736	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 2B 10 C2 0729 E1 56 3E 60 A2 C2 073E 16 2A C2 073E 16 2A C2 073E 16 2A C2 006E 7E 80 77 23 11 F2 0735 CD 006B 3A 3FF1 B7 C8 CD 075E	SNDLNE: LENDLP: SNDLP: SNCNTL:	MVI CALL PUSH LXI DAD MVI RAL JNZ DCX DCR JNZ DCX DCR JNZ POP MUV ANA JNZ CALL LDA RZ CALL	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M OBOH M, A H E SNDLP SNDCR LOCLM A	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry ; until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination ;local? ;remote, no time delay ;do delay</pre>	
071D 0722 0723 0724 0727 0729 0729 0720 0726 0730 0731 0734 0735 0736 0736 0736 0736 0736 0736 0737 0736 0736	CD 0174 E5 11 004F 19 16 40 7E 7E 17 AA C2 0734 2B 10 C2 0729 E1 56 3E 60 A2 0729 E1 56 3E 60 A2 006E 7E EE 80 77 23 10 F2 0735 CD 006B 3A 3FF1 B7 C8 CD 075E 15	SNDLNE: LENDLP: SNDLP: SNCNTL: CRACE:	MVI CALH LXI DAD MVI MUVI MUVI MAL XRA JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 77 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M OBOh M, A H E SNDLP SNDCR LOCLM A SDLY D	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space? . clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read charater ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ;until end of line ;send cr for termination ;local? ;remote.no time delay</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0726 0726 0730 0731 0734 0735 0736 0736 0735 0736 0737 0736 0738 0736 0738 0736 0736 0737 0744 0744 0744 0745 0744 0745 0746 0755 0755	CD 0174 E5 11 004F 17 7E 17 AA C2 0734 2B 10 C2 0729 E1 56 3E 60 A2 C2 073E 16 C2 073E 16 C2 073E 16 C2 073E 16 C2 0735 CD 006B 3A 3FF1 B7 C9 CD 075E 15 C2 0752	SNDLNE: LENDLP: SNDLP: SNCNTL: CRACE:	MVI CALL PUSH LXI DAD MVI MUI XRA JNZ DCR JNZ DCR JNZ DCR JNZ CALL DCR ZRI UCR JCR ZCR JNZ CALL DCR ZCR JNZ	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D, "*" WTACED A, M OBOH M, A H E SNDLP SNDCR LOCLM A SNDLP SNDCR LOCLM A	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination ;local? ;remote, no time delay ;d was ODh</pre>	
071D 071F 0722 0723 0724 0727 0729 0724 0728 0726 0730 0731 0731 0734 0735 0736 0736 0736 0737 0736 0738 0739 0732 0744 0742 0744 0745 0744 0745 0746 0747 0745 0755 0755	CD 0174 E5 11 004F 17 7E 7E 7E 7E 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C 7C	SNDLNE: LENDLP: SNDLP: SNCNTL: CRACE:	MVI CALL PUSH LXI DAD MVI MUVI MUVI XRA JNZ DCX JNZ POPV MVI ANA JNZ DCX JNZ POPV MVI CALL JNZ CALL LDA ORA QRZ CALL DCR JNZ CALL DCR JNZ DCX INX DCX INX DCX INX DCX DCX DCX DCX DCX DCX DCX DCX DCX DC	B, 0 CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D SNCNTL D, "*" WTACED A, M OBOh M, A H E SNDLP SNDCR LOCLM A \$DLY D \$D2 BELPRT	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination ;local? ;remote.no time delay ;d was ODh ;delay done, ring bell</pre>	
071D 0722 0723 0724 0727 0729 0724 0728 0726 0730 0731 0734 0735 0736 0736 0736 0738 0736 0736 0738 0736 0738 0736 0736 0737 0741 0742 0742 0744 0745 0745 0745 0759 0759	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 28 10 C2 0729 E1 56 3E 60 A2 C2 0735 CD 004E 7E EE 80 77 72 23 10 F2 0735 CD 004B 3A 3FF1 B7 C8 CD 075E 15 C2 0752 D3 01 C3 0443	SNDLNE: LENDLP: SNDLP: SNCNTL: CRACE: #D3:	MVI CALLH LXI DAD MVI MAU XRA JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D, "*" WTACED A, M OBOh M, A H E SNDCR LOCLM A SDLY D \$D3 BELPRT DCUR1	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;mask out msb ;space? , clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination ;local? ;remote, no time delay ;d was ODh</pre>	
071D 071F 0722 0723 0726 0727 0729 0724 0728 0726 0727 0728 0726 0730 0731 0734 0735 0736 0736 0739 0737 0736 0739 0736 0739 0736 0738 0739 0736 0737 0744 0744 0745 0744 0745 0751 0755 0755 0755 0755	CD 0174 E5 11 004F 17 A A C2 0734 28 C2 0734 28 C2 0739 E5 S6 C2 0729 E1 56 C2 0729 E1 6 C2 0729 C2 0729 C2 0735 C2 0735 C3 C1 006E 77 C3 C2 0735 C1 006B 3A 3FF1 B7 C9 CD 075E 15 C2 0752 D3 01 C3 0463 E5	SNDLNE: LENDLP: SNDLP: SNCNTL: CRACE:	MVI CALL DAD MVI MOVI MAL XRA DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, 0 CURLOC H D, 79 D, 040h A, M D LNSP H E ENDLP H D, M A, 060h D, 060h D, 070H D, 07	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination ;local? ;remote.no time delay ;d was ODh ;delay done, ring bell</pre>	
071D 0722 0723 0724 0727 0729 0724 0728 0726 0730 0731 0734 0735 0736 0736 0736 0738 0736 0736 0738 0736 0738 0736 0736 0737 0741 0742 0742 0744 0745 0745 0745 0759 0759	CD 0174 E5 11 004F 19 16 40 7E 17 AA C2 0734 28 10 C2 0729 E1 56 3E 60 A2 C2 0735 CD 004E 7E EE 80 77 72 23 10 F2 0735 CD 004B 3A 3FF1 B7 C8 CD 075E 15 C2 0752 D3 01 C3 0443	SNDLNE: LENDLP: SNDLP: SNCNTL: CRACE: #D3:	MVI CALLH LXI DAD MVI MAU XRA JNZ DCX DCX DCX DCX DCX DCX DCX DCX DCX DCX	B, O CURLOC H D, 79 D, 040h A, M D LNSP H E LENDLP H D, M A, 060h D, "*" WTACED A, M OBOh M, A H E SNDCR LOCLM A SDLY D \$D3 BELPRT DCUR1	<pre>;set cursor to ist loc ;get row start ;find row end ;hl=row end, clear carry ;rotated space ;read char ;space?, clear carry ;until find a non-space ;repeat loop ;hl=row start again ;read character ;screen all cntl codes ;not control code ;cntl code, send "*" instead ;write to ace ;read char again ;invert attribute ;store back ;next character ; until end of line ;send cr for termination ;local? ;remote.no time delay ;d was ODh ;delay done, ring bell</pre>	TL/F/5869-2

0760	CD 07F0		CALL	LUBD	;lookup baud constant	
0763	3E 1F		MVI	A,01Fh	;chg delay at high bauds	
0765	83		ORA	E .		
0766	5F	\$D1:	MOV CALL	E, A \$D2		
0767 076A	CD 059B 1B	\$D1;	DCX	su₂ D		
0768	15		DCR	a		
0760	14		INR	ă		
076D	F2 0767		JP	\$D1		
0770	D1		POP	D		
0771	E1		POP	H		
0772	C9		RET			
		PAGE				
	MACRO-ASSEME	LER V2. 0		PAGE	28	
CRT801						
0773					N (LOCAL)	
0773	CD 05E7	LRDCUR:		RDY	;y coordinate	
0776	2A 3FFC	CRECON.	LHLD	LROW	;get print out loc	
0779	5E		MOV	E, M	· · · · · · · · · · · · · · · · · · ·	
077A	23		INX	н		
077B	56		MOV	D, M		
0770	1 B		DCX	D	;lrow last location	
077D	12		STAX	D	;put to screen	
077E	CD 05E0		CALL	RDX	x coordinate	
0781	1B		DCX	D		
0782	12		STAX	D	put to screen	
0783	C9		RET			
		SCALE	(1004)			
0784	21 3FFE	SCALE:	LXI	H, FROW		
0787	3A 3FFA	SCALE.	LDA	CROW		
078A	BE		CMP	M	; frow=crow	
078B	CC 048C		CZ	ROLDWN	; if equ, roll down	
078E	2A 3FFA		LHLD	CROW	;get print out loc	
0791	28 SFFA		DCX	H	get print obt itt	
0792	2B		DCX	н		
0793	CD 0682		CALL	\$1		
0796	EB		XCHG	**	;hl=loc	
0797	1E 31		MVI	E, "1"		
0799	16 B1	SCLLP1:		D, "1"+0		
079B	BA		MVI	A, "9"+(981 h	
079D	72	SCLLP2:		M, D		
079E	14		INR INX	D	;"1" to "9"	
079F	23		CMP	н Н	next location exceeding "9"?	
0740	BA CD 070D			D SCLLP2	,exceeding 7	
07A1 07A4	C2 079D 3E 39		JNZ MVI	A, "9"		
0744	73		MOV	M, E		
0747	10		INR	E	;"1" to "8"	
	23		INX	H	; next location	
07A8 07A9	BB		CMP	E	; exceeding "8"?	
07A9 07AA	65 C2 0799		JNZ	SCLLP1	vexceeding of	
07AD	C2 0799 C9		RET	JULL I		
	-					
			AL INTER		NT I NUE ********	
07AE	2B	VTSUB:	DCX	******* H	; VCALEN	
07AF	74		MOV	м, н	<pre>interview in the interview in the interview intervi</pre>	
0780	2A 3FFB		LHLD	CUR	get cursor	
0783	36 03		MVI	M, 3	; write to DP8350	
0785	2A 3FF6		LHLD	TOP	; top of page	
0788	36 02		MVI	M, 2	; write to DP8350	
07BA	D3 40		OUT		;clr vert intr flip/flop	
0780	E1		POP	H	· · · · · · · · · · · · · · · · · · ·	
07BD	FB		EI			
07BE	C9		RET			
		PAGE				
STARPLEX	MACRO-ASSEME	LER V2. 0		PAGE	29	
CRT801						
07BF						
			ION} ace		line feed, upper/lower case	
07BF	DB 40	ACESW	IN	SETSW	read switch settings	
0701	E6 10		ANI	010h	-	
07C3	32 3FF2		STA	AULF	;set auto lf flag	
0706	DB 40		IN	SETSW	pread switch again	
	E6 20		ANI	020h	;read switch again	
0708						
07C8 07CA	C3 07D2		JMP	STUL	;set u/l case flag	

07CD	3A 3FF0		INIT A	CE, TOGGLE	ULCASE	(LOCAL)		
07D0	EE 20			XRI	020h	;toggle u	/l case	
0702	32 3FF0	5	TUL	STA	ULCASE	.		
07D5	CD 07F0			CALL	LUBD	,	aud constant	
07D8 07DB	21 9003 74			LXI MOV	H, ACECTL M, H	;set DLAB		
07DC	2E 01			MVI	L, 1			
07DE 07DF	72 28			MOV DCX	м, D Н	;set baud	high	
07E0	73			MOV		;set baud	low	
07E1 07E3	2E 03 36 02			MVI MVI	L,3 M,2	;7 bit,1	stop bit	
07E5	3A 3FF1	-	D A A C	LDA		;local?		
07E8 07E9	3C 21 9000	Ł	DACE:	INR LXI	A H, ACEDTA	•		
07EC 07ED	5E 23			MOV INX		;remove a ;ACEITR ma		
07EE	77			MOV			le ace intr	
07EF	69			RET				
07F0 07F2	DB 40 E6 OE	L	UBD:	IN ANI	SETSW OEh	;lookup b;	aud constant	
07F4	C6 F0	в	ADDR:	ADI	low B110);add base	addr	
07F6 07F7	6F 26 01			MOV MVI	L, A H, 1			
07F9 07FA	5E 23			MOV INX	E, M H	;get baud	low	
07FB	56			MOV		;get baud	high	
07FC	C9			RET END	START			
	EX MACRO-	-ASSEMBL	ER V2. (o	PAGE	30		
CRT901 Macros:								
Symbols								
\$1 \$5	0682 04A9	\$2 \$D1	05A0 0767	\$3 \$D2	0681 059B	\$4 \$D3	04A7 0752	
\$DLY	075E	\$G1	06AD	\$G2	06AC	\$G3	06B2	
ACE ACEITR	0038 9001	ACECTL ACESTU	9003 9005	ACEDTA ACESW	9000 07BF	ACEDUP ADDCUR	0018 05A2	
ATGUL	07CD	AULF	3FF2	B110	01F0	B1200	01F4	
B19200 B600	01FC 01F2	B2400 B9600	01F6 01FA	B38400 BADDR	01FE 07F4	B4800 BELL	01F8 0434	
BELPRT BSRTN	0001 0417	BGNDF C1D	04F7 023A	BS C7E	041F 0240	BS1 CALCX	0422 05A6	
CALCY	05B4	CALJMP	0150	CARRIN	0389	СВ	0204	
CE CG	020A 020E	CFB CHAR	0642 0372	CFBDIF CHKGM	0653	CFBLP CJMP	064A 0158	
CLRROW	OGDE	CLRSCN	04CE	CLRWLF	04E1	CMRTN	0250	
CPLP CRACE	056F 074A	CPYCTL CRLR	3FE7 0590	CQ CROW	0222 3FFA	CR CROWH	000D JFFB	
CTLRW1 CUREN	04E6 3FEC	CTLRW2 CURH	04E9 3FF9	CU CURLOO	022A 0174	CUR CURTMR	3FF8 3FEB	
CURULK	04AE	CX4FD	05B3	CY17D	05BE	DCROW	OBDB	
DCUR1 DICMD	0463 03CE	DEGRPH DICUR	06BC 001B	DEL DMYROW	0419 J 3F00	DFCLOC DRCFL	0171 067A	
DRLLP	0407	DRTLN	0400	DRTPG	04BA	DRWZCU	058C	
DWNCUR F10	0460 84D8	EDACE F11	07E8 85F3	ENGRPH F12	06B7 8472	F1 F13	0A5F 848C	
F14 F18	84AE 84F7	F15 F19	8DD1 8368	F16 F2	ADA2 0389	F17 F20	84F1 8358	
F21	8505	F22	8512	F23	AD26	F24	9E8A	
F25 F29	0419 8642	F26 F3	83DB 03B1	F27 F30	84CO 84CE	F28 F31	84BA 8CB6	
F32 F36	801B 86BC	F33 F4	871D 0C34	F34 F5	87BF 060A	F35 F6	0AC5 0437	
F7	041F	F8	844E	F9	8460	FBG	3F20	
FCHR2 FFSTRT	3780 3F50	FFCHK FFWCT	0267 3FEA	FFEND	3F9F 3FE6	FFRD FGNDF	3FE5 04F1	
FIFACE	004E	FRCR	0596	FROW	3FFE	FROWH	3FFF	
FS GSYMBL	0437 3FF3	FUNC HOMCUR	0168 04D8	GECNTL ICMD	3FEF 3FE8	GRAPH ILCHAR	068A 03AB	
ILELIN INSCHR	02C2 038D	ILNERG INSLNE	0547 0505	INIT INSLP	00E2 039B	INSCHA IRWOS	036B 05C1	
ISLRG	0526	ISNPA	053C	JMPADE	016A	KBD	0050	
KBDACE KBLK	01E4 04B6	KBDINT KLCRTN	02C9 0260	KBDL.CH KNACT\		KBDPRT KULCDE	0080 0003	
LCL	02F2	LCLFUN	005F	LCLINE	0002	LDFGD	04FA	
LEADIN LFEED	3FE9 03B1	LENDLP LFSCR	0729 03D2	LF LGPARA		LFD LIN	0301 8000	
LINC LINMDE	007E 0283	LINEXE LINP	02BE 3FA3	LINF LINPFN	3FA0 1 02A4	LINFH LINPRA	3FA1	TI /E/5060 05
ETHURE	5200	- 1 IAF	5, 1 5	L 1 (1) C P	. VERT	EAGE INFI		TL/F/5869-25

LINSET	0205	LINWCT	3FA2	LNSP	0734	LOCLM	3FF1
LRDCUR	0773	LROW	3FFC	LROWH	3FFD	LSTCHR	0300
LUBD	07F0	MIDCHR	0387	MOVDWN	054B	MOVROW	0550
NLC	0800	NLCSE	02EC	NOWRAP	0066	NPAGE	05F3
NPLP	05F9	NPLT	0601	NRW	3FF5	ONLINE	0345
STARPLI CRT801	EX MACRO	-ASSEMBLI	ER V2.0		PAGE	31	
OUTACE	0079	OVRNG	0320	P32SYM	0700	P38SYM	06F3
P4SP	06FB	P6SYM	06F7	PATTN	0112	PGM	04D5
PGMLP	0702	PSPC	0344	PTNLP	0117	PUSHSP	OBEC
RDCUR	05D1	RDFIFO	026E	RDNEQ	0444	RDX	05E0
RDY	05E7	RFFRNG	027A	ROLDWN	0480	ROLUP	0472
ROM2	034F	ROSEOD	0500	ROSFFU	OSCB	ROW	0008
ROW1	0082	ROW10	0094	ROW11	0096	ROW12	0098
ROW13	009A	ROW14	0090	ROW15	009E	ROW16	00A0
ROW17	00A2	ROW18	0044	ROW19	0046	ROW2	0084
ROW20	0048	ROW21	00044	ROW22	OOAC	ROW23	OOAE
ROW24	0080	ROW25	0082	ROW26	0084	ROW27	0086
ROW28	0000	ROW29	OOBA	ROW3	0086	ROW30	OOBC
ROW31	OOBE	ROW32	0000	ROW33	0002	ROW34	0004
ROW31 ROW35	0006	ROW36	0000	ROW37	00CA	ROW38	0000
ROW35	OOCE	ROW4	00088	RDW40	0000	ROW41	0002
ROW42	0002	ROW43	0006	ROW44	0008	ROW45	OODA
ROW46	OODC	ROW47	OODE	ROW47D	007E	ROW48	00E0
ROW48D	0080	ROW5	008A	ROW6	0072	ROW7	008E
ROWADD	0090	ROW9	0092	ROWDP	0028	ROWPRT	0040
RR1	0182	RR10	0194	RR11	0196	RR12	0198
RR13	019A	RR14	0190	RR15	019E	RR16	01A0
RR13	014A	RR19	0144	RR19	0146	RR1D	01E2
RR2	0184	RR20	0148	RR21	0144	RR22	01AC
RR23	0184 01AE	RR24	0180	RR25	0182	RR26	01B4
RR27	0186	RR28	0188	RR29	01BA	RR3	0184
RR30	01BC	RR31	OIBE	RR32	0100	RR33	0102
RR34	0104	RR35	0106	RR36	0108	RR37	01C2
			OICE	RR4	0188	RR40	0100
RR38 RR41	01CC 01D2	RR39 RR42	0102	RR43	0106	RR44	0100
		RR46	01DC	RR43		RR47D	017E
RR45	01DA	RR48D	0180	RR5	01DE 018A	RR47D	017E
RR48 RR7	01E0 018E	RR8	0190	RR9	0192	RTECTL	3FED
		RUADD	0700	RUNEQ	0192	RWRG	0060
RTN	025F		0667	SCLLP1	0799	SCLLP2	0790
SCALE	0784	SCATT	0518	SNCNTL	073E	SNDCR	006B
SETSW	0040	SLNERG		SPC		START	0000
SNDLNE	071D	SNDLP	0735	STOFCH	0020		0512
STFIFO	0244	STK	3FE5 06D0	STSP	0358 06CB	STOFLN STUL	07D2
	035F	STSCN	0033	TBLJMP	0200	TDAOSP	0611
TAB	060A	TABSTP					
TGLCL	0333	TMCUR	062F	TOP	3FF6	TOPH	3FF7
TSANSP	0620	TSATT	0635	UCUR1	0451	ULCASE	3FF0
UPCUR	044E	VCAL	011F	VCALEN	3FF4	VERPRT	0040
VERT	0010	VERTDP	0030	VRWRAP	0061	VTSUB	07AE
W1	1000	W2	2000	WAIT	014E	WFFRNG	025A
WTACEA	006D	WTACED	006E	ZROCUR	04DE		

No Fatal error(s)

TL/F/5869-26

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

 Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

N)	National Semiconductor Corporation	National Semiconductor Europe	National Semiconductor Hong Kong Ltd.	National Semiconductor Japan Ltd.
	1111 West Bardin Road	Fax: (+49) 0-180-530 85 86	13th Floor, Straight Block,	Tel: 81-043-299-2309
	Arlington, TX 76017	Email: cnjwge@tevm2.nsc.com	Ocean Centre, 5 Canton Rd.	Fax: 81-043-299-2408
	Tel: 1(800) 272-9959	Deutsch Tel: (+49) 0-180-530 85 85	Tsimshatsui, Kowloon	
	Fax: 1(800) 737-7018	English Tel: (+49) 0-180-532 78 32	Hong Kong	
		Français Tel: (+49) 0-180-532 93 58	Tel: (852) 2737-1600	
		Italiano Tel: (+49) 0-180-534 16 80	Fax: (852) 2736-9960	

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.