HP 91200B
TV Interface Kit

Programming and Operating Manual

HEWLETT-PACKARD COMPANY
11000 WOLFE ROAD, CUPERTINO, CALIFORNIA, 95014

PART NO. 91200-90006

Printed in U.S.A. APR 1977

PREFACE

This programming and operating manual provides the information required to program the HP 91200B TV Interface Kit. The HP 91200B Kit operates in a real-time software system and enables the user to display information on a user-supplied black-and-white (BW) or color television (TV) monitor.

Users of this manual should be proficient software programmers either in HP FORTRAN, HP Assembly Language, or HP RTE BASIC and must be familiar with the programming and operating procedures given in one of the following manuals:

- *HP Real-Time Executive-II Software System Programming and Operating Manual,* part no. 92001-93001.
- *HP Real-Time Executive-III Software System Programming and Operating Manual,* part no. 92060-90004.
- *HP Real-Time Executive Core-Based Software System Programming and Operating Manual,* part no. 29101-93001.
- *HP Real-Time Executive BASIC Software System Programming and Operating Manual,* part no. 29102-93001.
- *Multi-User Real-Time BASIC Reference Manual,* part no. 92060-90016.

For the user coding his TV interface kit application program in HP Assembly Language, required reading is the reference manual, Assembler, part no. 24307-90014. If the user is coding his application program in HP FORTRAN IV, he should procure the reference manual, HP FORTRAN IV, part no. 5951-1321. HP RTE BASIC is covered completely in the HP RTE BASIC system manuals listed above.

CONTENTS

ILLUSTRATIONS

TABLES

GENERAL INFORMATION

SECTION I

1-1. DESCRIPTION

The Hewlett-Packard Model 91200B TV Interface Kit processes digital computer information for display on a standard black-and-white (BW) or color television (TV) monitor. The HP 91200B generates a composite video signal and is selectively compatible with either American or European standard broadcast TV scan rates. In addition, the HP 91200B can supply non-standard scan rates to optimize its operation for TV monitors operating with 60 Hz vertical rates. Real-Time Executive (RTE) software (Option 422) for the HP 91200B Kit consists of the following items:

a. RTE Driver DVA13 Binary Tape, part no. 91200-16001.

b. Driver DVA13 Manual, part no. 91200-90005.

c. TV Interface Library Tape, part no. 91200-16002.

d. TV Interface Verification Tape, part no. 91200-16004.

e. This manual, part no. 91200-90006.

The TV interface kit driver (DVA13) and the library subroutines are used together to allow the user to program display information for one or more TV monitors. The driver is generally not accessed directly by the calling program; rather calls are made to the library subroutines which format the proper calling sequences to the driver. The TV display is refreshed by random access memory (RAM) on the TV kit's interface card; therefore compute and input/output (I/O) time are required only to change the display.

Several TV monitors may be connected in parallel to one TV interface card to display the same information on all monitors. Also, the TV kit's software is capable of operating one or more TV interface cards at the same time with each card supplying different information. If desired two or three TV cards can be operated jointly to supply gray-scale video to a black-and-white monitor or color video to a color monitor. Refer to the *HP 91200B TV Interface Kit Installation and Service Manual,* part no. 91200-90001, for interface kit hardware details.

1-2. OPERATING ENVIRONMENT

The operating environment for the TV interface kit must be:

a. An HP 2100A/S or 21MX Computer.

b. An HP RTE-II, RTE-III, RTE-C, or RTE-B Operating System.

Note: Hereafter in this manual, the term "RTE" will be used in place of "RTE-II, RTE-III, RTE-C, or RTE-B".

1-3. TV MONITOR CHARACTERISTICS

The TV monitor characteristics required for operation with the TV interface kit are discussed in the *HP 91200B TV Interface Kit Installation and Service Manual,* part no. 91200-90001.

OPERATION

SECTION II

2-1. INTRODUCTION

This section provides information on the hardware-controlled operating modes of the TV interface kit and includes the verification procedures for checking operation of the kit. Actual operation of the HP 91200B is discussed in Section III of this manual.

2-2. OPERATING MODES

The user may select any one of three modes of operation as described in the *HP 91200B TV Interface Kit Installation and Service Manual,* part no. 91200-90001. The required operating mode is determined by the type of TV monitor used with the TV kit and must be selected before the TV interface card in the kit is installed in the computer. The three modes are as follows:

- European broadcast standard TV scan rates. The TV interface kit utilizes a 256- by 256-point matrix for information display.
- American broadcast standard TV scan rates. The display matrix is reduced vertically from 256 points to 240 (horizontal resolution remains at 256 points).
- Non-standard TV scan rates. This mode provides the full 256- by 256-point matrix for TV monitors operating with 60 Hz vertical rates.

In addition to the black-and-white (BW) video generated by a single TV interface card, the user can generate gray-scale or color video by operating two or three TV cards in a multi-card configuration. Special installation instructions for the three modes of operation described above and for multi-card operation are provided in the TV interface kit installation and service manual.

2-3. VERIFICATION PROCEDURES

The verification program provides a convenient means of checking the TV monitor's adjustments and of verifying operation of the TV interface kit in the RTE operating system environment. (The program requires less than 5000 words of background area.) To use the verification program, proceed as follows:

Note: The verification program cannot be used in RTE BASIC systems.

a. With the RTE system generated and operational, load the TV verification tape (part no. 91200-16004) into the photoreader.

Note: Information pertaining to RTE system generation is contained in Section III of the *RTE Driver DVA13 Programming and Operating Manual,* part no. 91200-90005.

b. On the terminal keyboard, enter the RTE operator command:

ON,LOADR

c. The system loads the tape and outputs (on the terminal) the message:

/LOADR:LOAD

d. Enter the command:

GO,LOADR,1

e. The system outputs the message:

/LOADR:TVERF READY
LOADR:$END

f. Run the verification program by entering the command:

ON,TVERF,LU

where: LU is the logical unit number of the terminal to be used for the remainder of this procedure. If your system is configured for multiple terminal operation, LU defaults to the terminal you are using if omitted; otherwise, the default is to the system terminal.

The system outputs the message:

TV INTERFACE LU=

g. Enter the logical unit number of the TV interface card to be verified. The system outputs the question:

IS CARD IN AMERICAN SCAN MODE?

h. Enter YES or NO as appropriate for the scan mode used by the card. (If you make a typing error, the question will be repeated.) The system then outputs the question:

IS THIS A 1 CARD SYSTEM?

i. Enter YES for a one-card (BW) system and NO for a multi-card (color or multi-level gray) system. The system outputs the question:

DO YOU KNOW THE COMMANDS?

j. Enter YES or NO. If you enter NO, the system outputs a listing of the verification command codes as shown in figure 2-1 (one-card system) or figure 2-2 (multi-card system) followed by the question: COMMAND? If you enter YES, the system outputs COMMAND? without listing the command codes. (An entry other than YES or NO will cause a repeat of the question DO YOU KNOW THE COMMANDS?)

k. Enter one of the valid command codes. If any code other than those listed is entered, the program will ask: DO YOU KNOW THE COMMANDS?

l. If GA, CR, SE, PO, LO, or DE is entered in response to COMMAND? in a multi-card system, the program asks the question: WHAT COLOR?

m. Enter the single-character color code, as listed in figure 2-2, for the desired color or gray-scale level.

n. Enter the DE command and observe the display it presents. This display provides the quickest way of being sure that you made the correct entry in step h above.

Note the following items:

- If the card is operating in the European standard or non-standard scan mode, and if you entered NO in step h, the display should be as shown in figure 2-3a.
- If the card is in the American standard scan mode, and if you entered YES, the display should be similar to that shown in figure 2-3b.
- If the card's scan mode does not correspond to the answer entered in step h, you will see either the loss of the top line or a message stating that you are in the wrong mode.

After the DE command or any other command except EX is executed, the system outputs a request for another command. The commands are defined in figures 2-1 and 2-2; you can use any or all of them as desired to verify operation of the TV interface kit and to check the adjustments of the TV monitor. Note the following:

- Each displayed pattern remains on the screen until replaced by another one or erased by the ER command.
- If you make a typing error when entering a command, the system repeats the question: DO YOU KNOW THE COMMANDS? The system responds as described in step j.
- Figure 2-4a shows the pattern obtained by the GA command when the TV interface card is operating in either the European standard or non-standard scan mode; figure 2-4b shows the corresponding pattern obtained when operating in the American standard scan mode.
- The crosshatch pattern (CR command) shows 16 spaces both left and right of a vertical double line marking the center of the screen. For the American standard scan mode, there are 15 spaces above and below a horizontal double center line (see figure 2-5b); for the other scan modes, there are 16 spaces above and below this center line (see figure 2-5a). If the card's scan mode does not correspond with the answer entered in step h, the number of spaces above the center line will be different from the number of spaces below it. These differences are shown in figure 2-5.
- The SE command generates squares in all four corners of the display, with the squares appearing as rectangles due to the 4:3 aspect ratio of the screen. If the TV card is in the American standard scan mode and the program is not, the squares in the upper corners of the screen will not be visible.
- The PO command, which illuminates all points, will also indicate the lack of dc restoration or high voltage regulation in the TV monitor. Additionally, by following the PO command with a command that displays another pattern, you can check the monitor's ability to sync to the composite video signal. If the monitor temporarily loses sync when the pattern change occurs, consider using the TV card's external sync line for the monitor.
- The LO command presents a facsimile of the Hewlett-Packard logo with the words "TV INTERFACE CARD" appearing above it.
- The DE command generates a number of calls whereby all of the modules in the video display library are used. The resulting displays for the European standard or non-standard scan mode and the American standard scan mode are shown in figure 2-3.
- The ER command totally erases the screen. The screen will be black if normal video polarity is selected and white if inverse video polarity is selected.
- The IN command changes video polarity. The program begins in the normal polarity and each execution of IN switches the polarity.
- The FL command causes any pattern to flash (switch video polarity rapidly back and forth) for a preset period of time at a preset rate. When FL is entered, the program requests the pattern to be flashed:

  PATTERN TO FLASH?

  When a valid command (GA, CR, SE, PO, LO, or BA) is entered, the pattern is drawn and then the polarity is reversed. For multi-card systems, the program asks for the display color:

WHAT COLOR?

When a color code is entered, the program fills the screen with a background in the complementary color, followed by the pattern in the specified color. Color complements are as follows:

2-Card Color System	3-Card Color System
Yellow — Black	White — Black
Red — Green	Red — Cyan
	Green — Magenta
	Blue — Yellow

In a one-card system, or when white is specified in a color system, the flashing rate is twice as fast as that for other colors. The flashing continues for a few seconds and then stops.

Entering a non-valid command (ER, IN, LU, or MO) after FL causes the flashing command to be ignored and the requested command to be executed.

- The BA command, available only for multi-card systems, displays vertical bars on the screen, each bar showing a system color. The displays for a two-card system and a three-card system are shown in figure 2-6. The patterns can be used for adjusting colors on the monitor, or for setting the contrast, pedestal, and offset controls on a multilevel gray system. These adjustments are described in the *HP 91200B TV Interface Kit Installation and Service Manual,* part no. 91200-90001. For flashing no complementary color is used. The BA pattern writes over any previous pattern.

- The LU command causes a sequential request for logical unit number, scan mode, number of cards, and command.

- The MO command causes a request for scan mode, number of cards, and command.

- The EX command terminates the verification program; the system outputs the message:

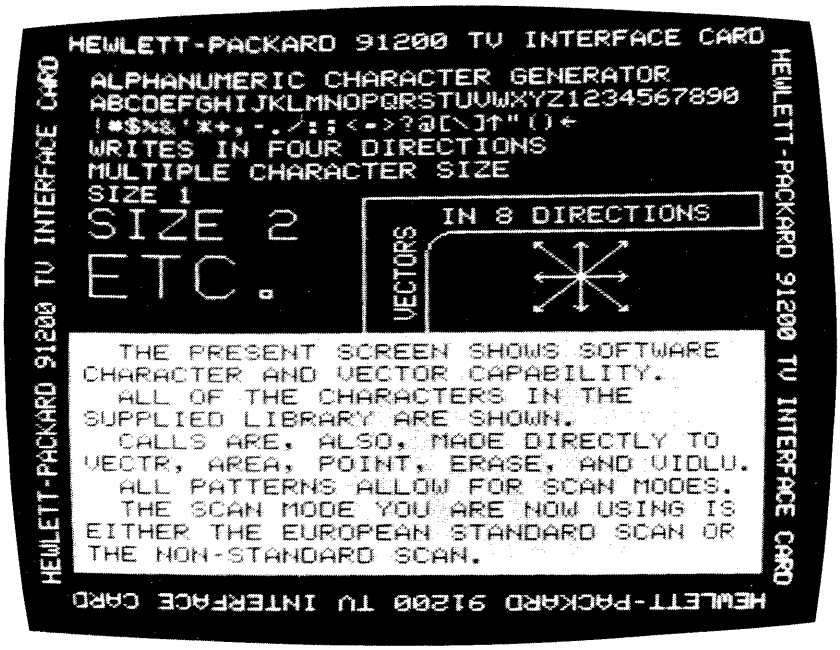
  TVERF:STOP 0077

```
COMMANDS ARE ENTERED AS FOLLOWS:
    GA  GAIN PATTERN FOR ADJUSTING MONITOR HEIGHT, WIDTH,
          POSITION, AND FOCUS.
    CR  CROSSHATCH PATTERN FOR ADJUSTING MONITOR
          LINEARITY AND PINCUSHION.
    SE  RECTANGLES IN THE FOUR CORNERS OF THE SCREEN
          FOR CHECKING DISPLAY SETTLING TIME EFFECTS.
    PO  WRITES ALL POINTS ON THE SCREEN TO CHECK
          FOR MISSING POINTS.
    LO  WRITES HP LOGO ON THE SCREEN.
    DE  DEMONSTRATES ALL FEATURES OF DISPLAY LIBRARY.
    ER  ERASES THE SCREEN.
    IN  INVERT THE VIDEO POLARITY ON THE SCREEN.
    FL  FLASH ANY SPECIFIED PATTERN.
    LU  TO ENTER A NEW TV LU.
    MO  TO ENTER A DIFFERENT SCAN MODE.
    EX  EXITS THIS PROGRAM.
```

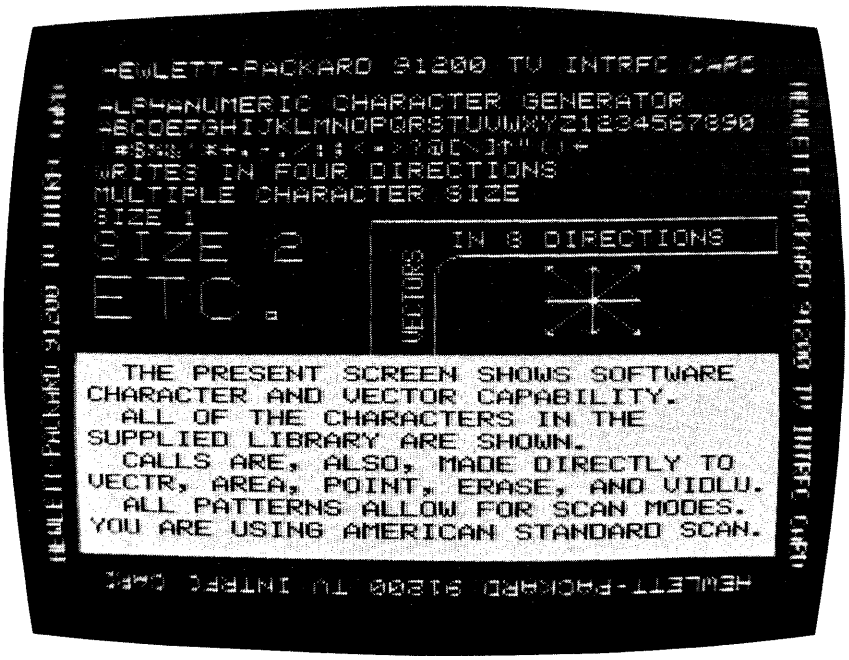
Figure 2-1. Verification Program Commands — One Card System

```
COMMANDS ARE ENTERED AS FOLLOWS:
    GA  GAIN PATTERN FOR ADJUSTING MONITOR HEIGHT, WIDTH,
          POSITION, AND FOCUS.
    CR  CROSSHATCH PATTERN FOR ADJUSTING MONITOR
          LINEARITY AND PINCUSHION.
    SE  RECTANGLES IN THE FOUR CORNERS OF THE SCREEN
          FOR CHECKING DISPLAY SETTLING TIME EFFECTS.
    PO  WRITES ALL POINTS ON THE SCREEN TO CHECK
          FOR MISSING POINTS.
    LO  WRITES HP LOGO ON THE SCREEN.
    DE  DEMONSTRATES ALL FEATURES OF DISPLAY LIBRARY.
    ER  ERASES THE SCREEN.
    IN  INVERT THE VIDEO POLARITY ON THE SCREEN.
    FL  FLASH ANY SPECIFIED PATTERN.
    BA  COLOR OR GRAY SCALE BARS.
    LU  TO ENTER A NEW TV LU.
    MO  TO ENTER A DIFFERENT SCAN MODE.
    EX  EXITS THIS PROGRAM.


 THE COLOR CODES ARE AS FOLLOWS:

 CODE  COLOR     GRAY LEVEL
 ----  -----     ----------
   B   BLUE          1/8
   G   GREEN         1/4
   C   CYAN          3/8
   R   RED           1/2
   M   MAGENTA       5/8
   Y   YELLOW        3/4
   W   WHITE         7/8
```

Figure 2-2. Verification Program Commands — Multi-Card System

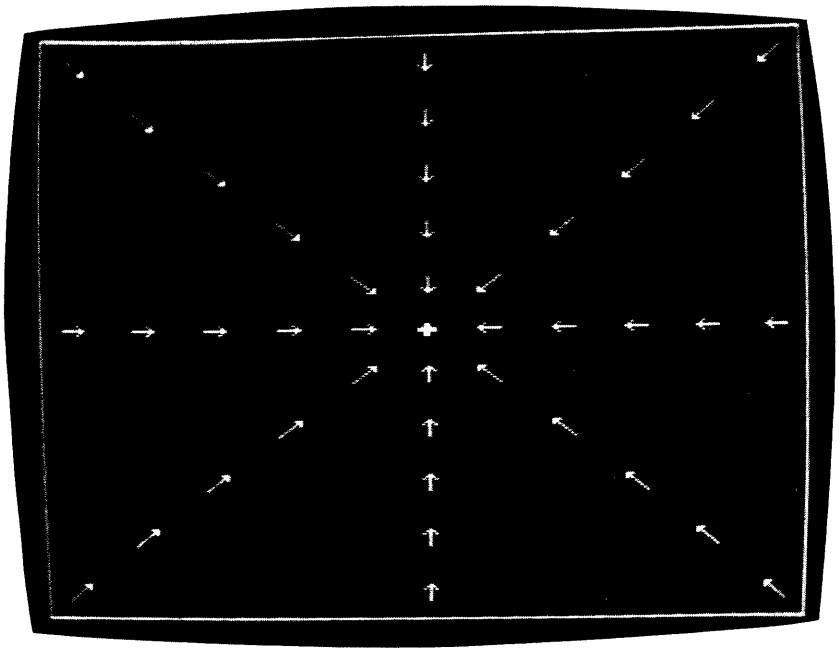


a. European Standard or Non-Standard Scan Mode

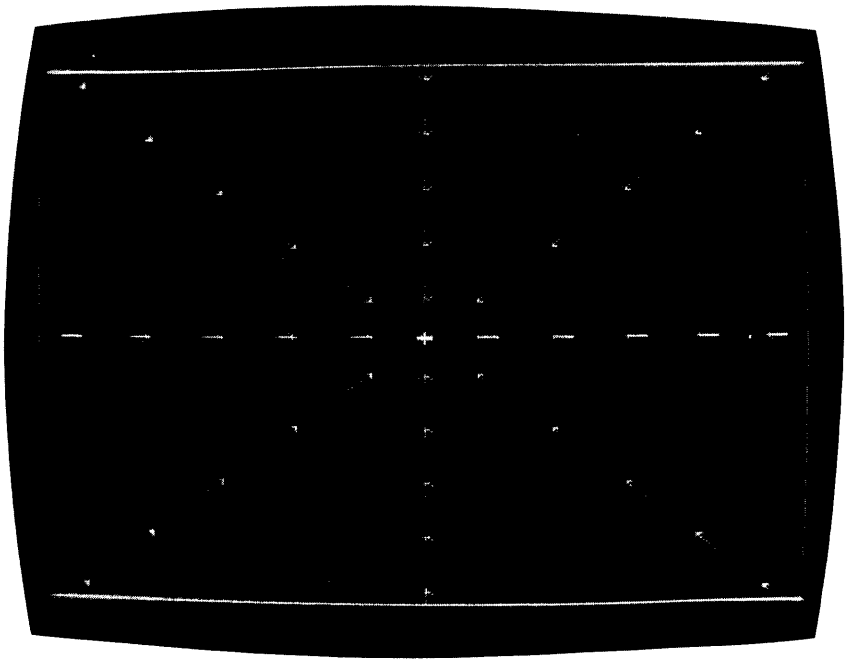


b. American Standard Scan Mode

Figure 2-3. DE Command Patterns



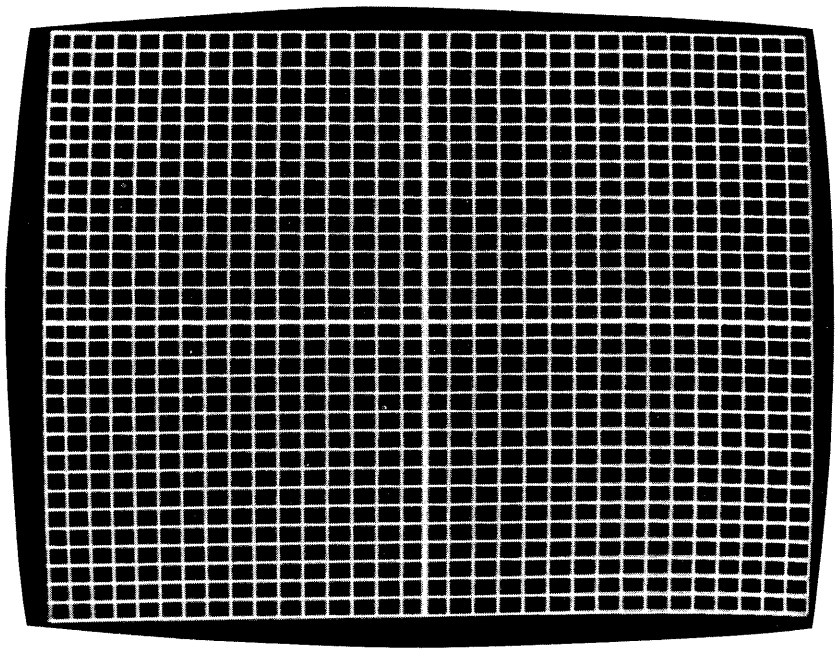
a. European Standard or Non-Standard Scan Mode



b. American Standard Scan Mode

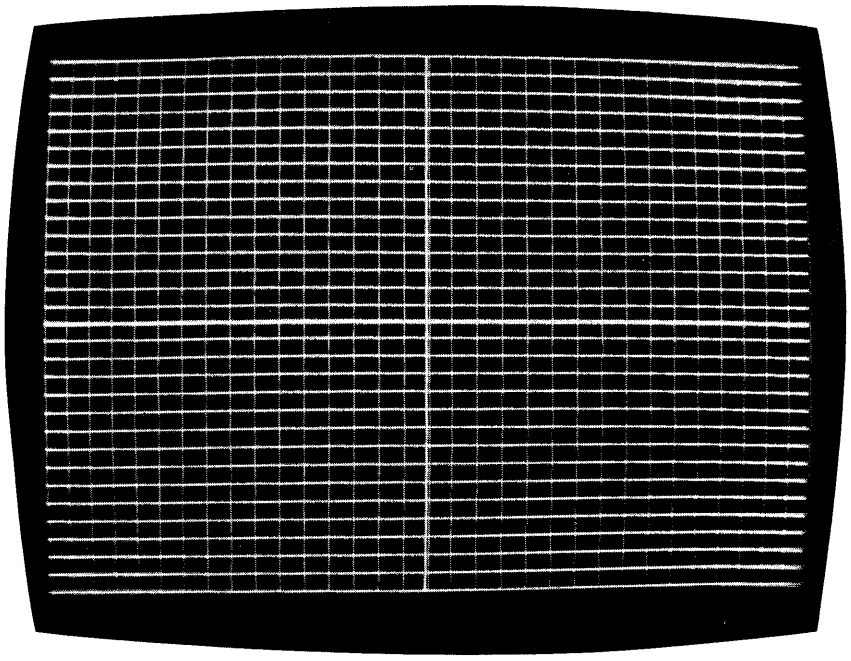
Figure 2-4. GA Command Patterns

FOURTEEN SPACES ARE VISIBLE ABOVE CENTER LINE IF CARD IS IN AMERICAN STANDARD SCAN MODE.



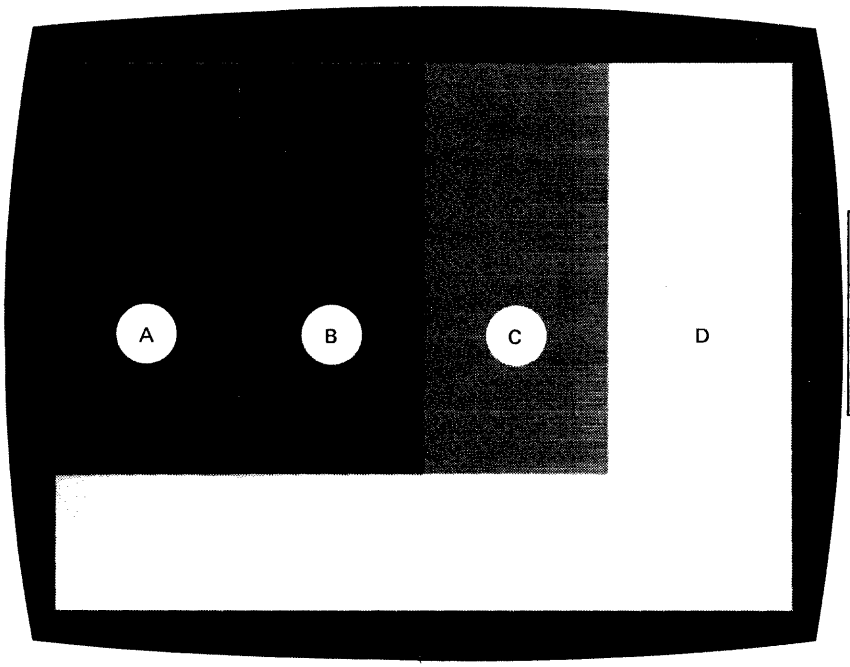
a. European Standard or Non-Standard Scan Mode

SEVENTEEN SPACES ARE VISIBLE ABOVE CENTER LINE IF CARD IS IN EUROPEAN STANDARD OR NON–STANDARD SCAN MODE.



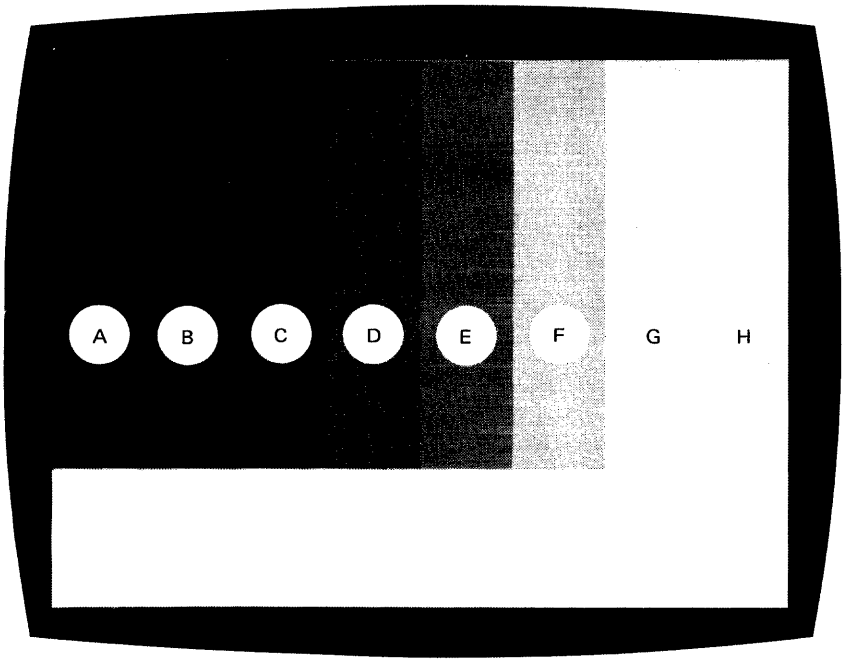
b. American Standard Scan Mode

Figure 2-5. CR Command Patterns



BAR	COLOR	GRAY-SCALE SHADE
A	BLACK	BLACK
B	GREEN	1/4
C	RED	1/2
D	YELLOW	3/4 (WHITE)

a. Two-Card Color or Multi-Level Gray System



BAR	COLOR	GRAY-SCALE SHADE
A	BLACK	BLACK
B	BLUE	1/8
C	GREEN	1/4
D	CYAN	3/8
E	RED	1/2
F	MAGENTA	5/8
G	YELLOW	3/4
H	WHITE	7/8 (WHITE)

b. Three-Card Color or Multi-Level Gray System

Figure 2-6. BA Command Patterns

PROGRAMMING

SECTION III

3-1. INTRODUCTION

This section provides information on programming the TV interface kit by using the library subroutines supplied with the kit. Also included are overall considerations of display programming. Character set modification and generation is discussed in Appendix A.

3-2. SOFTWARE

The TV kit's software includes a driver and a library having nine subroutines plus a character set module. Generally, all display programming is done with the subroutines briefly described below:

SUBROUTINE	DESCRIPTION
VIDLU	Specifies logical unit and video polarity; must be called first.
ERASE	Completely erases the display screen.
VECTR	Draws programmable length vectors in any of eight directions.
CHAR	Displays a string of ASCII characters, starting at specified point in display.
CHARS	Interfaces HP BASIC to the CHAR routine.
CODE	Allows FORTRAN write using the Formatter.*
VAREA	Illuminates or darkens any specified rectangular screen area, or selectively erases an area without erasing entire screen.
POINT	Displays specified points.
PNTS	Converts HP BASIC floating point values into integer values for display by POINT.
VEND	Retrieves vector end-point coordinates of the last vector or character.

* Subroutine CODE is not part of the TV kit library but is part of the RTE/DOS Relocatable Library (either EAU or Floating Point version).

3-3. SUBROUTINE CALLS

The subroutine calls used to program the TV kit are defined in the following paragraphs. Call formats for HP FORTRAN, HP BASIC, and HP Assembly programming languages are shown. Note in the FORTRAN calls that brackets ([ ]) enclose optional parameters which always assume the specified default values. If any optional parameter is used, those preceding it must be used also.

Note: All parameters are type integer.

3-4. VIDLU

Purpose:

Designates the logical unit number of the TV monitor on which information will be displayed and sets the video polarity of the display.

Formats:

FORTRAN

```
CALL VIDLU (LU[,ISENS])
```

BASIC

```
CALL VIDLU (L1[,S1])
```

Both parameters must be specified.

Note: L1 = LU S1 = ISENS

Assembly

```
EXT VIDLU
  .
  .
  .
JSB VIDLU
DEF *+2 (or 3)
DEF LU
DEF ISENS     optional parameter
⟨return point⟩
```

Where:

LU
the logical unit number of the TV monitor.

ISENS
video polarity indicator

ISENS = 0
normal video polarity. Display will be in color set by MODE selector. (See VECTR, CHAR/CHARS, AREA, or POINT/PNTS description.) Colors available for single- and multi-card systems are as follows:

1-Card System	2-Card System	3-Card System	
White	Yellow	White	Yellow
Black	Red	Red	Magenta
	Green	Green	Cyan
	Black	Blue	Black

INSENS ≠ 0
inverse video polarity. Causes color set by MODE to complement. Complementary color pairs are as follows:

1-Card System	2-Card System
White — Black	Yellow — Black
	Red — Green

3-Card System

White — Black
Red — Cyan
Green — Magenta
Blue — Yellow

ISENS
defaults to 0 if omitted.

Comments:

Since the other TV kit calls do not include a parameter that specifies logical unit (LU), a VIDLU call must be used before any of the other TV kit calls in your display program or subprogram. Once the LU and video polarity are set they remain set until changed by a subsequent VIDLU call.

Repeated calls to VIDLU with the same logical unit number and alternating sense parameters may be employed to flash attention to information on the screen without changing it.

3-5. ERASE

Purpose:

Performs a complete screen erase to the polarity previously set by call to VIDLU. This call also clears the power fail indicator (see Section 3-17). Normal video polarity causes all points on the screen to be darkened, inverse video polarity results in all points being illuminated.

Formats:

FORTRAN

```
CALL ERASE
```

BASIC

```
CALL ERASE
```

Assembly

```
EXT ERASE
  .
  .
  .
JSB ERASE
DEF *+1
⟨return point⟩
```

3-6. VECTR

Purpose:

Displays a line (vector) by writing a sequence of points on the display screen.

Formats:

FORTRAN

```
CALL VECTR (IX,IY,IDIRC,LEN[ ,ITYPE[ ,MODE
      [ ,LODU[ ,IPF]]]])
```

BASIC

```
CALL VECTR (X,Y,D,L,[ ,T[ ,M[ ,D1[ ,P]]]])
```

All parameters must be specified.

Note: X = IX Y = IY D = IDIRC
L = LEN T = ITYPE M = MODE
D1 = LODU P = IPF

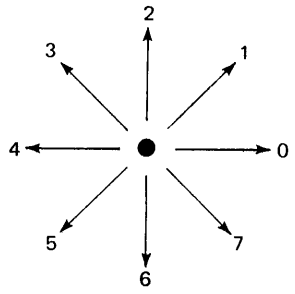
Assembly

```
EXT VECTR
   .
   .
   .
JSB VECTR
DEF *+5+n      n = number of optional parameters
DEF IX
DEF IY
DEF IDIRC
DEF LEN
DEF ITYPE  }
DEF MODE   }
DEF LODU   }   optional parameters
DEF IPF    }
⟨return point⟩
```

Where:

IX,IY
x and y coordinates of the starting point of the vector. If IX is negative, IY is ignored and the vector is started from the end of the previous vector. IX = IY = 0 defines the lower left corner of the display.

IDIRC
direction of the vector. The values of IDIRC are defined by the illustration shown below.



For values greater than 7 or less than 0, a proper direction is still achieved. For example: a value of +11 is the same as +3, and a value of −1 is the same as +7. VECTR accomplishes this conversion by always masking the IDIRC parameter as follows:

IDIRC = IAND (IDIRC,7)

LEN
the length of the vector in point spacings. If LEN is positive, the vector is drawn. If LEN is negative, the absolute value is used for the length and the vector is drawn 180° from the specified direction. A zero length vector is simply a point.

Note: Vectors drawn by VECTR do not automatically stop at the edge of the display matrix but may "continue" on the opposite side of the display. For example, if you program a vector to be drawn left-to-right from coordinates x = y = 0 and LEN = 382, the entire bottom line will be drawn plus about half of the second line (y = 1). LEN = −32768 should not be used; this value simply causes the point buffer (see LODU below) to be dumped while all other parameters in the call are ignored.

ITYPE
the type of vector to be drawn; defined as follows:

ITYPE = 0
non-drawing vector (space to new point).

ITYPE = 1
first and last points are suppressed. Used to connect between two existing points.

ITYPE = 2
all points are drawn.

ITYPE = 3
first point is suppressed. Used for continuation from an existing point.

ITYPE defaults to 2 if the last three parameters are omitted.

MODE
the color of points in call, defined as follows

MODE	3-CARD COLOR	3-CARD BW	2-CARD COLOR
0	White	7/8 (White)	Yellow
1	Black	Black	Black
2	Red	1/2	Red
3	Green	1/4	Green
4	Blue	1/8	Black
5	Yellow	3/4	Yellow
6	Magenta	5/8	Red
7	Cyan	3/8	Green

MODE	2-CARD BW	1-CARD BW
0	3/4 (White)	White
1	Black	Black
2	1/2	White
3	1/4	Black
4	Black	Black
5	3/4 (White)	White
6	1/2	White
7	1/4	Black

Note: Assumes ISENS = 0 (normal video polarity). If ISENS ≠ 0 (inverse video polarity), colors complement. (See VIDLU description.)

MODE
defaults to 0 if the last two or three parameters are omitted.

LODU
the point buffer load/dump indicator.

LODU = 0
load internal point buffer (part of supplied software).

LODU ≠ 0 dump point buffer.

LODU defaults to 0 if omitted.

IPF
If used returns 0 if no power fail has occurred since last **ERASE** call. Returns a 1 if a power fail has occurred (see Section 3-17).

Comments:

The LODU parameter allows you to minimize I/O operations when making a series of VECTR calls. The calls load the internal buffer with several vectors; the last call in the series dumps the buffer for display. The buffer can hold 350 words (points); if it becomes completely full during the series of VECTR calls, program execution is interrupted automatically, the buffer is dumped, and then execution continues.

The vectors drawn by VECTR are approximated by a sequence of points on a 256- by 256-point matrix. The length of a vector is programmed as a number of point spacings which are longer diagonally than horizontally and vertically. Due to the 4:3 aspect ratio of the TV monitor, both vertical and diagonal vectors are "shortened" relative to horizontal vectors. Thus a right triangle drawn with vertical, horizontal, and diagonal vectors specified to be n spacings long is displayed with the vertical vector relatively equal to 0.75n, the horizontal equal to 1.0n, and the diagonal equal to 1.25n (not 1.414n). To display an optical square, a program would have to specify a vertical-to-horizontal vector ratio of 1.333 to 1.

3-7. CHAR/CHARS

Purpose:

Displays characters specified by a buffer passed to it by the user's calling program.

Formats:

FORTRAN

```
CALL CHAR (IX,IY,IBUFR[,ISIZE[,IDIRC
          [,NUMBR[,MODE[,IPF]]]])
```

BASIC

```
CALL CHARS (X,Y,"character string"
          [,S[,D[,O[,M[,P]]]]])
```

or

```
CALL CHARS (X,Y,A$[,S[,D[,O[,M[,P]]]]])
```

All parameters must be specified.

Note: X = IX Y = IY A$ = IBUFR
S = ISIZE D = IDIRC 0 = NUMBR
M = MODE P = IPF

Assembly

```
EXT CHAR
 .
 .
 .
JSB CHAR
DEF *+4+n   n = number of optional parameters
DEF IX
DEF IY
DEF IBUFR
DEF ISIZE
DEF IDIRC   }
DEF NUMBR   }
DEF MODE    } optional parameters
DEF IPF     }
⟨return mode⟩
```

Where:

IX,IY
the x and y coordinates of the character strings' starting point (lower left corner of the first character when IDIRC = 0).

IBUFR
an integer array containing packed ASCII characters. The array may be constructed in any of the following ways:

1. Make the first word of the array the number (positive integer) of characters to be displayed. The second and subsequent words contain the packed character string. NUMBR must be set to zero. This is the way strings are constructed in HP BASIC.

2. Pack the entire buffer with characters to be displayed; define NUMBR as the number of characters.

3. Pack the buffer with ASCII characters; enclose the characters to be displayed in parentheses. Parentheses may be part of the display (inside the delimiter paren-

theses) as long as they are nested. NUMBR must be a negative value. This method is most useful with CODE and FORTRAN format statements.

ISIZE
the character size. If equal to 1, then the standard 5- by 7-point matrix is used. For values greater than 1, the vectors making up the character are multiplied in length by ISIZE. (See table 3-1.) The call will be rejected if ISIZE is negative. ISIZE defaults to 1 if specified as 0 or if the last four parameters are omitted.

IDIRC
a number that determines the orientation of the displayed characters. (See figure 3-1).

IDIRC = 0 — normal left-to-right writing.

IDIRC = 1 — bottom-to-top writing.

IDIRC = 2 — upside-down, right-to-left writing.

IDIRC = 3 — top-to-bottom writing.

IDIRC
defaults to 0 if the last three or four parameters are omitted.

Note: CHAR masks the IDIRC parameter to achieve a proper direction for values greater than 3 or less than 0. Refer to the description of VECTR.

NUMBR
the number of characters in the string when only characters to be displayed are in the string array. If the first word in the array specifies the number of characters, NUMBR must be 0. If the string is delimited by parentheses, NUMBR must be any negative value. NUMBR defaults to 0 if the last two, three, or four parameters are omitted.

MODE
the color of points in call, defined as follows:

MODE	3-CARD COLOR	3-CARD BW	2-CARD COLOR
0	White	7/8 (White)	Yellow
1	Black	Black	Black
2	Red	1/2	Red
3	Green	1/4	Green
4	Blue	1/8	Black
5	Yellow	3/4	Yellow
6	Magenta	5/8	Red
7	Cyan	3/8	Green

MODE	2-CARD BW	1-CARD BW
0	3/4 (White)	White
1	Black	Black
2	1/2	White
3	1/4	Black
4	Black	Black
5	3/4 (White)	White
6	1/2	White
7	1/4	Black

Note: Assumes ISENS = 0 (normal video polarity). If ISENS ≠ 0 (inverse video polarity), colors complement. (See VIDLU description.)

MODE
defaults to 0 if the last one, two, three, or four parameters are omitted.

IPF
If used returns 0 if no power fail has occurred since last **ERASE** call. Returns a 1 if a power fail has occurred (see Section 3-17).

Comments:

A character string displayed by CHAR does not automatically stop at the edge of the display matrix but may continue on the opposite side of the screen. If this occurs, a character may be "split" on the matrix's edges and there will be no "line feed". Therefore, you should write your programs to control the placement of each line of characters.

Characters are formed by drawing a sequence of vectors. Subroutine CHAR assumes that a buffer passed to it contains a packed ASCII string. For each character in the string, CHAR retrieves a sequence of vector codes from the library character set module called [TABL and translates them into parameters for calls to VECTR. CHAR can multiply the length of each vector by an integer parameter to display various size characters, and can rotate each vector to display characters along any of the four 90° axes selected by the direction parameter. As discussed in Appendix A, the ASCII string codes may be replaced by other byte codes and the [TABL module may be modified to display user-designed characters.

3-8. CODE

Purpose:

Gives FORTRAN programs the capability of using the Formatter to perform internal ASCII-to-binary and binary-to-ASCII conversions within memory through READ and WRITE statements. (Assembly and BASIC programs do not require the use of CODE.)

Format:

FORTRAN

CALL CODE

Table 3-1. ISIZE Parameter

ISIZE =	CHARACTER SIZE (POINTS)		POINTS BETWEEN CONSECUTIVE CHARACTERS	MAXIMUM CHARACTERS PER LINE
	WIDTH	HEIGHT		
1	5	7	1	42
2	9	13	3	21
3	13	19	5	14
4	17	25	(2 × ISIZE) − 1	10

Note: The character matrix is:

[(4 x ISIZE) + 1] x [(6 x ISIZE) + 1]

The largest practical value for ISIZE is 42, which provides a matrix of 169- by 253-points. ISIZE values greater than 42 present incomplete display characters.

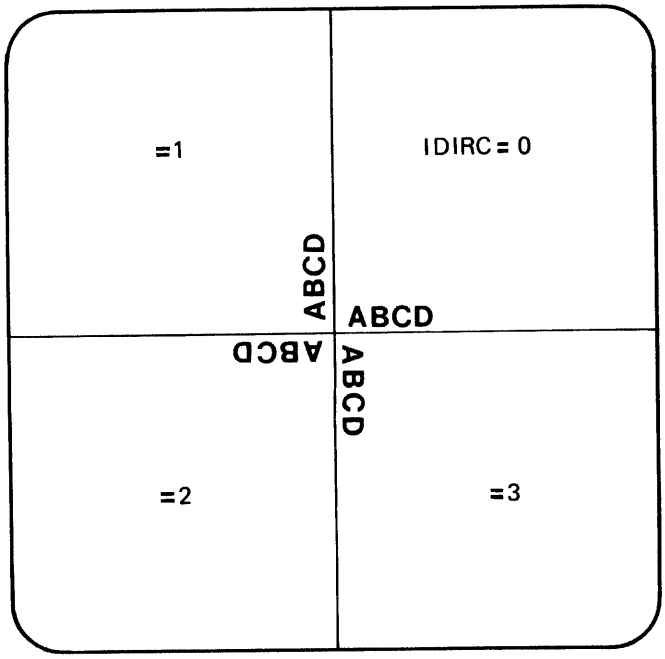


Figure 3-1. IDIRC Parameter for CHAR/CHARS and VAREA

Comments:

For the TV interface kit, Relocatable Library* subroutine CODE may be used in conjunction with CHAR and a WRITE statement to output ASCII character displays.

*Library subroutines are described in the Relocatable Subroutines Reference Manual (part no. 02116-91780).

The CODE call statement must immediately precede the WRITE statement. The two statements are logically considered as a single statement and, therefore, if a label is used it should be attached to the CODE call statement. When using CODE to generate an ASCII buffer that will be passed to CHAR, the parentheses required by CHAR must be inserted in the FORMAT statement as ASCII data; the FORMAT statement's parentheses are not sufficient. The example statement sequence shown below illustrates the use of CODE.

```
      X = 2.*6.+5.*4.

      CALL CODE
      WRITE(IBUFR,10)X
10    CALL CHAR(20,222,IBUFR,3,0,-1)
      FORMAT ("(EXAMPLE: X=",F6.2,")")
```

The displayed result is:

```
EXAMPLE: X= 32.00
```

3-9. VAREA

Purpose:

Writes (or erases) a specified rectangular area of the display screen.

Formats:

FORTRAN

```
CALL VAREA (IX,IY,IDX,IDY[,IDIRC[,MODE[,
            IPF]]])
```

BASIC

```
CALL VAREA (X,Y,X1,Y1,[,D[,M[,P]]])
```

All parameters must be specified.

Note: X = IX Y = IY X1 = IDX
Y1 = IDY D = IDIRC M = MODE
P = IPF

Assembly

```
EXT VAREA
  .
  .
  .
JSB VAREA
DEF *+5+n   n = number of optional parameters
DEF IX
DEF IY
DEF IDX
DEF IDY
DEF IDIRC  }
DEF IMODE  } optional parameters
DEF IPF    }
⟨return point⟩
```

Where:

IX,IY
the x and y coordinates of the rectangular area's starting point (lower left corner of area when IDIRC = 0).

IDX,IDY
The x and y lengths, in point spacings, of the rectangular area (assuming IDIRC = 0). If either or both are < 0, the call will be ignored.

Note: Areas drawn by VAREA do not automatically stop at the edge of the display matrix but may continue on the opposite side of the display.

IDIRC
a number that determines the orientation of the area from its starting point. IDIRC is defined for VAREA the same as it is for CHAR (see figure 3-1) and defaults to 0 if the last two parameters are omitted.

MODE
the color of points in call, defined as follows:

MODE	3-CARD COLOR	3-CARD BW	2-CARD COLOR
0	White	7/8 (White)	Yellow
1	Black	Black	Black
2	Red	1/2	Red
3	Green	1/4	Green
4	Blue	1/8	Black
5	Yellow	3/4	Yellow
6	Magenta	5/8	Red
7	Cyan	3/8	Green

MODE	2-CARD BW	1-CARD BW
0	3/4 (White)	White
1	Black	Black
2	1/2	White
3	1/4	Black
4	Black	Black
5	3/4 (White)	White
6	1/2	White
7	1/4	Black

Note: Assumes ISENS = 0 (normal video polarity). If ISENS ≠ 0 (inverse video polarity), colors complement. (See VIDLU description.)

MODE
defaults to 1 if the last one or two parameters are omitted.

IPF
If used returns 0 if no power fail has occurred since last **ERASE** call. Returns a 1 if a power fail has occurred (see Section 3-17).

Comments:

VAREA allows a specified rectangular area of the screen to be selectively written or erased. Note that since the dimensions of an area are defined in terms of point spacings, there will always be one more row and one more column of points than the value of the IDX and IDY parameters. For example, if IDX = IDY = 2, a 3- by 3-point area will be displayed.

VAREA may be used to erase messages or data values from an area before a new message or value is displayed in that area. With the exception of comma (,) and semicolon (;), all size-1 characters are coded to fit within a 5- by 7-point matrix. Both "," and ";" extend below the normal bottom lines of points by one point spacing. When VAREA is used to erase a normal left-to-right line of text which may include commas or semicolons, the IY parameter of the VAREA call should be ISIZE less than IY of the corresponding CHAR call (in which ISIZE is specified) and IDY should equal character height plus ISIZE. (Refer to table 3-1.)

A calling program may use VAREA to "reverse" the video polarity of part of the screen. For example, if the last VIDLU call has set the video polarity to normal (white on black), part of the screen can be used for writing black on white. This is done by using AREA in the normal polarity mode to illuminate an area, and then writing the information into the white area (e.g., displaying with CHAR in black). In two-card and three-card systems similar presentations can be achieved in complementary colors.

3-10. POINT/PNTS

Purpose:

Displays the points stored as packed yx coordinates in a buffer.

Formats:

FORTRAN

CALL POINT (IBUFR[,NUMBR[,MODE[,IPF]]])

BASIC

CALL PNTS(B[1][,N[,M[,P]]]) . . . . . RTE-B

CALL PNTS(B[,N[,M[,]]]) . . . . . .Multi-User Real-Time BASIC

All parameters must be specified.

Note: N = NUMBR M = MODE P = IPF

B differs from IBUFR:

$$\begin{aligned} B(1) &= X_1 \\ B(2) &= Y_1 \\ B(3) &= X_2 \\ B(4) &= Y_2 \\ . &= . \\ . &= . \\ B(2n-1) &= X_n \\ B(2n) &= Y_n \end{aligned}$$

NUMBR = n

Assembly

```
EXT POINT
  .
  .
  .
JSB POINT
DEF *+2+n   n = number of optional parameters
DEF IBUFR
DEF NUMBR  }
DEF MODE   } optional parameters
⟨return point⟩
```

Where:

IBUFR
an integer array containing point coordinates, with bits 15 through 8 of each word being the y coordinate and bits 7 through 0 being the x coordinate.

NUMBR
the number of points to be displayed from IBUFR. If NUMBR is 0, the call is ignored. NUMBR defaults to 1 if the last two parameters are omitted.

MODE
the color of points in call, defined as follows:

MODE	3-CARD COLOR	3-CARD BW	2-CARD COLOR
0	White	7/8 (White)	Yellow
1	Black	Black	Black
2	Red	1/2	Red
3	Green	1/4	Green
4	Blue	1/8	Black
5	Yellow	3/4	Yellow
6	Magenta	5/8	Red
7	Cyan	3/8	Green

MODE	2-CARD BW	1-CARD BW
0	3/4 (White)	White
1	Black	Black
2	1/2	White
3	1/4	Black
4	Black	Black
5	3/4 (White)	White
6	1/2	White
7	1/4	Black

Note: Assumes ISENS = 0 (normal video polarity). If ISENS≠ 0 (inverse video polarity), colors complement. (See VIDLU description.)

MODE
defaults to 0 if the last one or two parameters are omitted.

IPF
If used returns 0 if no power fail has occurred since last **ERASE** call. Returns a 1 if a power fail has occurred (see Section 3-17).

Comments:

It is recommended that the size of the buffer not exceed 5K words. This is especially important when there may be possible dual-channel port controller (DCPC) contention between the TV interface card and other very fast and possibly synchronous devices. If a larger buffer is required, it may be necessary to increase the time-out value chosen for the TV interface card at RTE system generation. Refer to Section III of the RTE Driver DVA13 Programming and Operating Manual for time-out programming instructions.

If the calling program computes x and y values for the point coordinates, the values for each point can be combined into one word as follows.

Note: This procedure is not necessary for programs in HP BASIC.

a. Insure that neither x nor y exceed 255 and that neither is negative.

b. Multiply y by integer 256 (in FORTRAN), or rotate y eight bits (ALF,ALF in Assembly).

c. Combine x and y with an inclusive OR:IBUFR(I) = IOR (IX,IY) in FORTRAN; or IOR in Assembly.

3-11. VEND

Purpose:

Retrieves the coordinates of the end point of the last vector drawn on the display.

Formats:

FORTRAN

```
CALL VEND (IX,IY)
```

BASIC

```
CALL VEND (X,Y)
```

Note: X = IX Y = IY

Assembly

```
EXT VEND
  .
  .
  .
JSB VEND
DEF *+3
DEF IX
DEF IY
⟨return point⟩
```

Where:

IX,IY
the x and y coordinate parameters to be used in a subsequent VECTR, CHAR, or VAREA call.

Comments:

CHAR completes each character with a non-writing vector to the point at which the next character should start. In the example statement sequence shown below, VEND is used to enable four characters from the second buffer to be written in line with six characters from the first.

```
  .
  .
CALL CHAR (20,212,IBUFR,1,0,6)
CALL VEND (IX,IY)
CALL CHAR (IX,IY,JBUFR,1,0,4)
```

3-12. DRIVER CALLS

The TV interface kit driver is normally accessed only through the TV kit subroutines. If desired, however, the driver may be accessed directly by the driver calls described in the *RTE Driver DVA13 Programming and Operating Manual,* part no. 91200-90005, supplied with TV interface kit Option 422.

3-13. DISPLAY CONSIDERATIONS

Consider the TV screen as a blank writing surface to be divided into lines or areas for displaying messages or data. In dividing the screen into lines for characters, note the following items:

a. With the 5- by 7-point character matrix of library module [TABL, the columns of points required by a line of n characters equals:

$([n-1]\times 6+5)\times$ ISIZE (Refer to table 3-1.)

b. If three point spacings are used between lines having size-1 characters, the maximum number of lines that can be displayed is 28.

c. If the TV interface card is set for American broadcast standard scan, the vertical resolution is reduced from 256 points to 240. (The maximum number of lines discussed in item b above becomes 26.)

When the TV interface kit is used with a data measurement system, the user may prepare a subroutine to display any of the measured data items. Figure 3-2 is an example subroutine which, for simplicity, displays only one data item. More sophisticated programs and subroutines can be prepared to display data on a conditional basis; for example, out-of-limits data, data related to a hardware interrupt, or data selected by name from the system console. Such a system requires corresponding common arrays for data names, values, and units of measurement.

Table 3-2. Card Selection for Gray-Scale Operation

MODE SELECTOR	SHADE	CARD A (1/2)	CARD B (1/4)	CARD C (1/8)
0	7/8 (White)	0	0	0
1	Black	1	1	1
2	1/2	0	1	1
3	1/4	1	0	1
4	1/8	1	1	0
5	3/4	0	0	1
6	5/8	0	1	0
7	3/8	1	0	0

Notes:

a. 0 = color present (on).
1 = color not present (off).

b. It is assumed that normal video polarity (ISENS = 0) is selected. If inverse video polarity (ISEN ≠ 0) is selected, the cards' on and off states are complemented. Thus, 7/8 (white) becomes black, black becomes 7/8 (white), 1/2 becomes 3/8, etc.

```
      SUBROUTINE DISP1(IDATA,IFLAG)
      DIMENSION IASCI(20)
C
C  SET REENTRY POINT.
C
      IF(IFLAG.NE.0)GO TO 10
C
C  SET LOGICAL UNIT TO 7; DEFAULT WRITING SENSE TO NORMAL.
C  SET SCREEN TO BLACK.
C
      CALL VIDLU(7)
      CALL ERASE
C
C  DISPLAY WORDS (17 CHARACTERS) FROM STATEMENT 2000 IN
C    POINT COLUMNS 20 THROUGH 120 USING WHITE.
C
      CALL CODE
      WRITE(IASCI,2000)
      CALL CHAR(20,222,IASCI,1,0,-1)
C
C  DISPLAY WORDS (SIX CHARACTERS) FROM STATEMENT 2010 IN
C    POINT COLUMNS 171 THROUGH 205 USING WHITE.
C
      CALL CODE
      WRITE(IASCI,2010)
      CALL CHAR(171,222,IASCI,1,0,-1)
C
C  CONVERT SIGNAL MEASURED BY CALLING PROGRAM TO VOLTS
C    AND THEN TO TEMPERATURE VALUE.
C
   10 VOLTS = 10.24*IDATA/32767
      TEMP = VOLTS/0.043
C
C  ESTABLISH COLOR: GREEN IF TEMP < 100 DEG. C;
C    YELLOW IF 100 <= TEMP < 500; AND RED IF TEMP >=500.
C
      MODE = 3
      IF(TEMP.GE.100)MODE = 5
      IF(TEMP.GE.500)MODE = 2
C
C  ERASE OLD TEMPERATURE VALUE AND DISPLAY NEW VALUE (SIX CHAR-
C    ACTERS) IN POINT COLUMNS 131 THROUGH 165 USING THE COLOR
C    SELECTED ABOVE.
C
      CALL VAREA(131,221,34,8,0)
      CALL CODE
      WRITE(IASCI,1000) TEMP
      CALL CHAR(131,222,IASCI,1,0,6,MODE)
C
 1000 FORMAT(F6.2)
 2000 FORMAT("(TURBINE INLET AIR)")
 2010 FORMAT("(DEG. C)")
      RETURN
      END
```

Figure 3-2. Display Subroutine Example: One Data Item

3-14. MULTI-CARD PROGRAMMING

The special considerations for operating two or three TV interface cards jointly to generate either gray-scale video or video signals for color TV are discussed in the following paragraphs.

3-15. GRAY-SCALE

Gray-scale video allows up to seven programmable shades of gray between 1/8-level and 7/8-level (white) to be displayed on the TV monitor. During hardware installation, the master card (card A) in a three card black-and-white system is allocated a gray-scale shade of 1/2, the first slave card (card B) a shade of 1/4, and the second slave card (card C) a shade of 1/8. The video outputs from all three cards are connected to amplitude summing circuitry on card A which outputs the summed video to the TV monitor. In operation, the MODE selector in a subroutine call simultaneously controls the video output from the three cards and the amplitude summation such that any gray-scale value between 1/8 and 7/8 can be automatically output to the monitor. Black is the absence of video from all three cards. The cards selected to program each of the gray-scale shades are listed in table 3-2. In a two-card black-and-white system, card A is allocated a gray-scale shade of 1/2 and card B a shade of 1/4. This provides three programmable shades between 1/4 and 3/4 plus black.

3-16. COLOR

Color video allows up to eight programmable colors (including black and white) to be displayed on a color TV monitor. During hardware installation, the master card (card A) in a three-card color system is allocated the color red, the first slave card (card B) the color green, and the second slave card (card C) the color blue. The video outputs from the three cards are connected to the appropriate video inputs on a RGB color monitor. In operation, the MODE selector in a subroutine call simultaneously selects the cards' outputs such that the monitor will "mix" the colors on the screen to produce the desired color. The cards selected to produce each of the available colors are listed in table 3-3.

Table 3-3. Card Selection for Color Operation

MODE SELECTOR	COLOR	CARD A (RED)	CARD B (GREEN)	CARD C (BLUE)	COMPLEMENTARY COLOR
0	White	0	0	0	Black
1	Black	1	1	1	White
2	Red	0	1	1	Cyan
3	Green	1	0	1	Magenta
4	Blue	1	1	0	Yellow
5	Yellow	0	0	1	Blue
6	Magenta	0	1	0	Green
7	Cyan	1	0	0	Red

Notes:

a. 0 = color present (on).
1 = color not present (off).

b. It is assumed that normal video polarity is selected (ISENS = 0). If inverse video polarity is selected (ISENS ≠ 0), the cards' on and off states are complemented. Thus, white becomes black, black becomes white, red becomes cyan, etc.

c. In a two-card color system, card C is omitted. Normal video polarity colors are red, green, yellow, and black. Complementary color pairs are red - green and yellow - black.

3-17. POWER FAIL AUTO RESTART

A power fail occurrance may be detected within an assembly, FORTRAN, or BASIC program by checking the optional parameter (IPF) in any subroutine call. The normal procedure for checking the 'IPF' parameter is to perform the particular subroutine call then perform:

a. An 'IF statement' in FORTRAN
b. An 'IF . . . THEN statement' in BASIC
c. A conditional branch in assembly

Use the following procedure to add the power fail feature to your RTE system:

Program 'AUTOR' must be modified to make the 'power fail detect' parameter effective.

STEP 1: Modify the fortran source 'AUTOR' by entering as the first executable statement:

CALL EXEC(3,600B+LUX)

where, 'LUX' is one logical unit pointing to a particular 'TV' equipment table entry.

For example: Suppose we have two TV's in our system. We therefore have two 'TV' equipment table entries. We also have 'N' logical units assigned to the first equipment table entry, and 'M' logical units assigned to the second equipment table entry. We simply place one of the 'N' logical units in our first "CALL EXEC (3,600B+LUN)" and one of the 'M' logical units in our second "CALL EXEC (3,600B+LUM)"

Notice that there is one "CALL EXEC (3,600B+LUX)" for each 'TV' equipment table entry in the system.

STEP 2: Modify source programs to include the 'power fail detect' parameter in calls to the TV library.

STEP 3: Check the status of the 'power fail' parameter at various locations in each program.

HOW IT WORKS

After a power fail, the first program to be executed will be 'AUTOR'. First, the TV screen is erased. The 'CALL EXEC(3,600B+LUX)' statement(s) which you have entered in 'AUTOR' will set the power fail detect parameter to 1. Next your user program which was running during the power failure is run from the current point of suspension. At various points in your user program you check the power fail detect parameter to see if a power failure has occurred.

Note: You should only include the power fail detect parameter in your subroutine call if it will later be checked. This will reduce overhead in your program.

See the example program in FORTRAN for more details (Figure 3-3).

```
&NUSNS T=00003 IS ON CR00200 USING 00012 BLKS R=1345

0001   FTN4,L
0002         PROGRAM NUSNS,2,32766
0003   C
0004   C=============================================
0005   C THIS PROGRAM PRINTS A MESSAGE ON THE TV AND
0006   C ASKS FOR A SYSTEM USER NAME. IT THEN UPDATES
0007   C A COUNT ON THE SCREEN AND CHECKS FOR POWER
0008   C FAIL. IF POWER FAIL HAS OCCURED "POWER FAIL!"
0009   C WILL BE PRINTED AND THE PROGRAM RESUMED.
0010   C
0011   C
0012   C  *ON,NUSNS,A,B,C
0013   C
0014   C    A = LU# OF TERMINAL
0015   C    B = LU# OF TV
0016   C    C = TEST POWER FAIL OUTPUT IF NONZERO
0017   C
0018   C=============================================
0019   C
0020         DIMENSION NAME(8),IPRM(5),IBFA(5),INUS(6),ISYS(4)
0021         DIMENSION IBFR(4),IPW(7),NAM1(7)
0022         EQUIVALENCE (IBFA(2),IBFR),(NAME(2),NAM1)
0023         DATA IBFA/8,2H  ,2H  ,2H  ,2H  /
0024         DATA ISYS/6,2HSY,2HST,2HEM/
0025         DATA NAME/14,2H  ,2H  ,2H  ,2H  ,2H  ,2H  ,2H  /
0026         DATA IPW/12,2HPO,2HWE,2HR ,2HFA,2HIL,2H! /
0027         DATA INUS/10,2HIN,2H U,2HSE,2H B,2HY /
0028         CALL RMPAR(IPRM)
0029         ILU=IPRM(1)
0030         IF(ILU.EQ.0) ILU=1
0031         ITV=IPRM(2)
0032         IF(ITV.EQ.0)ITV=12
0033         CALL VIDLU(ITV)
0034         CALL ERASE
0035         IF(IPRM(3).EQ.0) GO TO 85
0036   95    DO 65 IJ=1,10
0037         CALL VIDLU(12,1)
0038         CALL CHAR(10,102,IPW,3,0,0,2)
0039         CALL CHAR(11,103,IPW,3,0,0,2)
0040         CALL CHAR(12,104,IPW,3,0,0,2)
0041         CALL CHAR(13,105,IPW,3,0,0,2)
0042         DO 64 IJX=1,10000
0043   64    CONTINUE
0044         CALL VIDLU(12,0)
0045         DO 63 IJX=1,10000
0046   63    CONTINUE
0047         CALL ERASE
0048   65    CONTINUE
0049   85    CALL CHAR(10,102,INUS,3,0,0,2)
0050         CALL CHAR(11,103,INUS,3,0,0,2)
0051         CALL CHAR(12,104,INUS,3,0,0,2)
0052         CALL CHAR(13,105,INUS,3,0,0,2,IPF)
0053         IF(IPF.EQ.1) GO TO 95
```

Figure 3-3. Power Fail Example (Sheet 1 of 2)

```
0054          CALL CHAR(10,153,ISYS,3,0,0,2)
0055          CALL CHAR(11,154,ISYS,3,0,0,2)
0056          CALL CHAR(12,155,ISYS,3,0,0,2)
0057          CALL CHAR(13,156,ISYS,3,0,0,2,IPF)
0058          IF(IPF.EQ.1) GO TO 95




0059          WRITE(ILU,23)
0060    23    FORMAT(/"NUSNS:  ENTER NAME: ←")
0061          READ(ILU,24)NAM1
0062    24    FORMAT(7A2)
0063          CALL CHAR(10,51,NAME,3,0,0,2)
0064          CALL CHAR(11,52,NAME,3,0,0,2)
0065          CALL CHAR(12,53,NAME,3,0,0,2,IPF)
0066          IF(IPF.EQ.1) GO TO 95
0067          K=0
0068    35    CALL CODE
0069          WRITE(IBFR,12)K
0070          CALL CHAR(10,5,IBFA,4,0,0,2)
0071          CALL CHAR(11,6,IBFA,4,0,0,2)
0072          CALL CHAR(12,7,IBFA,4,0,0,2)
0073          CALL CHAR(13,8,IBFA,4,0,0,2,IPF)
0074          IF(IPF.EQ.1) GO TO 95
0075          DO 19 J=1,32000
0076    19    CONTINUE
0077          CALL CHAR(10,5,IBFA,4,0,0,1)
0078          CALL CHAR(11,6,IBFA,4,0,0,1)
0079          CALL CHAR(12,7,IBFA,4,0,0,1)
0080          CALL CHAR(13,8,IBFA,4,0,0,1,IPF)
0081          IF(IPF.EQ.1) GO TO 95
0082    12    FORMAT(I6)
0083          K=K+1
0084          GO TO 35
0085          END
0086          END$
```

Figure 3-3. Power Fail Example (Sheet 2 of 2)

CHARACTER SET MODIFICATION

APPENDIX A

A-1. INTRODUCTION

This appendix provides information on module [TABL of the TV interface kit software and includes a discussion of character design and modification of the character set module.

A-2. CHARACTER DISPLAY

Characters to be displayed by the TV kit are made with vectors that are specified by a packed string code passed to subroutine CHAR. (The string is usually, but not necessarily, ASCII.) CHAR checks each 8-bit byte code to see if it corresponds to a set of vector codes in module [TABL. If the byte code corresponds to a vector set, CHAR obtains the corresponding set from [TABL and uses it to generate calls to subroutine VECTR which displays the character. If correspondence does not exist for byte codes 140 through 177 octal (e.g., ASCII lower case), CHAR converts them to codes 100 through 137 (ASCII upper case) and repeats the check; if there is still no correspondence, the codes are passed over without further action. If there is no correspondence for codes 000 through 037 octal, the codes are simply passed over without action.

The [TABL module supplied with the TV kit contains vector code sets corresponding to byte codes 040 through 137 octal. If desired, the user can modify [TABL to increase the number of vector code sets by having the added sets correspond to byte codes less than 040 and/or greater than 137. [TABL can also be modified to replace several sets with code sets for different user-designed characters, or the entire character font can be replaced by vector code sets for a different font such as Japanese Katakana.

A-3. CHARACTER DESIGN

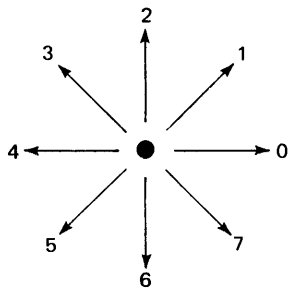
Each vector in a character is defined by a vector code contained in one byte; the code set for the character is packed two bytes per computer word. Each byte contains type, direction, and length codes in the following format:

TTDDDLLL (each letter represents one bit)

where:

T/type: 0 = non-drawing vector (space to new point).
1 = first and last points are suppressed.
2 = all points are drawn.
3 = first point is suppressed.

D/direction.
The direction of the vector. Direction values are defined by the following illustration.



L/length.
The length of the vector in point spacings (0 to 7).

In the TV kit's [TABL module, each size-1 character is designed for display within a 5- by 7-point matrix with two point spacings following each character. The maximum vector length that can be specified by a vector code is seven point spacings; this means that size-1 characters could be designed for any matrix up to eight-by-eight points (e.g., six-by-eight). Also, the matrix width may be reduced for narrow characters. In designing a character, note that the sequence in which vectors are drawn for the character is arbitrary but the following rules must be observed:

- The starting point of the character must be the lower left corner of the matrix.
- Each vector must start at the end of the previous vector.
- The last vector for each character must end at the starting point of the next character.

As a design guide for the user, Figure A-1 shows the vector codes and the resulting vectors that are used to display the letter "S".

In designing a character, consider the effect that the ISIZE parameter in subroutine CHAR will have on the character. You should design your characters so that they will look correct at any practical size. Taking shortcuts in designing characters may lead to size problems. For example, Figure A-2 shows two ways in which the letter "o" can be designed. The size-1 letters appear identical in both designs but when ISIZE=3 the "o" designed with 10 vectors looks much better than the one designed with 5 vectors.

Directory entry: ABS .S–.+070000B

- ABS .S– : relative position in vector word table
- +070000B : 7 words define the character

Packed vector code (octal) set for letter "S".

010671
141311
150731
161331
150711
141371
035063

TT:0 = non-drawing

1 = first and last points suppressed

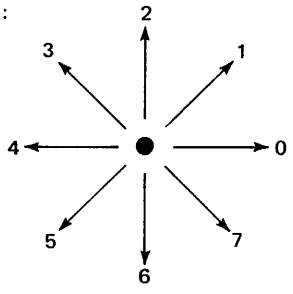
2 = all points drawn

3 = first point suppressed

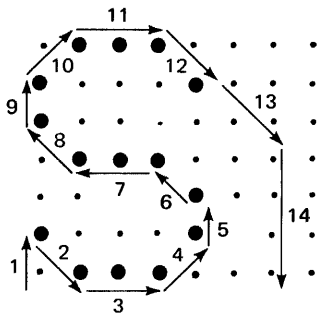
VECTOR BYTE CODES

VECTORS	FIRST BYTE								SECOND BYTE							
	T	T	D	D	D	L	L	L	T	T	D	D	D	L	L	L
1 and 2	0	0	0	1	0	0	0	1	1	0	1	1	1	0	0	1
	0		2			1			2		7			1		
3 and 4	1	1	0	0	0	0	1	0	1	1	0	0	1	0	0	1
	3		0			2			3		1			1		
5 and 6	1	1	0	1	0	0	0	1	1	1	0	1	1	0	0	1
	3		2			1			3		3			1		
7 and 8	1	1	1	0	0	0	1	0	1	1	0	1	1	0	0	1
	3		4			2			3		3			1		
9 and 10	1	1	0	1	0	0	0	1	1	1	0	0	1	0	0	1
	3		2			1			3		1			1		
11 and 12	1	1	0	0	0	0	1	0	1	1	1	1	1	0	0	1
	3		0			2			3		7			1		
13 and 14	0	0	1	1	1	0	1	0	0	0	1	1	0	0	1	1
	0		7			2			0		6			3		

DDD:



LLL: 0 to 7 point spacings.



Sequence and direction of vectors for the letter "S".

Figure A-1. Vectors for the Letter "S"

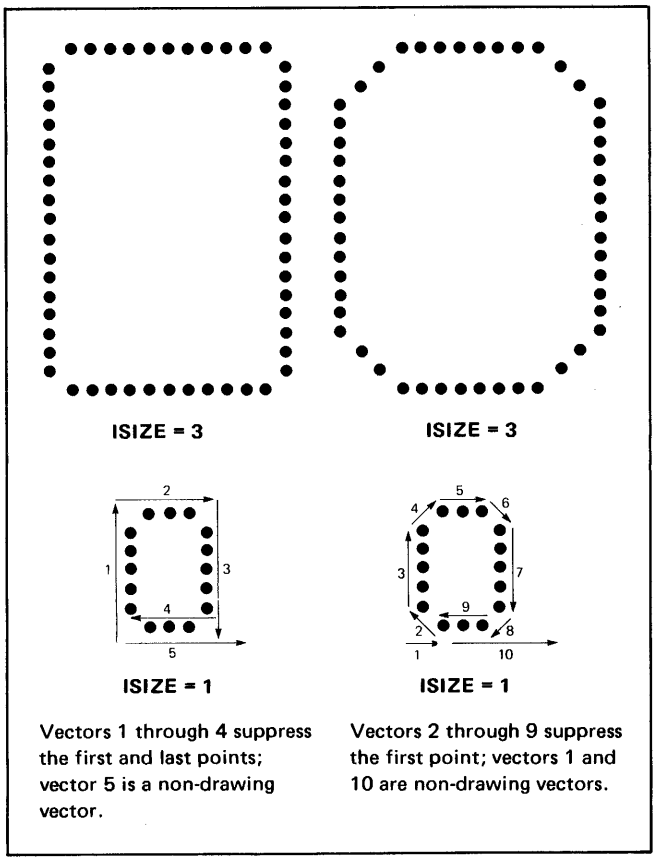


Figure A-2. Example Designs: Bad and Good

A-4. [TABL MODIFICATION

Figure A-3 is a source listing of the [TABL module supplied with the TV kit. (The library source tape, HP Part No. 91200-18002, can be ordered from your nearest Hewlett-Packard Sales and Service Office.) The module consists of two parts: a directory and a table. The table contains the vector code word sets for each character. The directory defines the following:

- The maximum number of characters that can be in the character set.
- The number of vector code words for each vector set.
- The relative position in the table of the first vector code word of each set.

The maximum number of characters in the set is determined by the octal values of MASK and BASE. When subroutine CHAR accesses [TABL, it uses MASK to exclude invalid byte codes and it subtracts BASE from the byte code value to find the position of the character entry in the directory. For example, the ASCII byte code for a space character is 040 octal and 040 minus 040 (the BASE) is 0; therefore, the space character is the first entry (defined by the statement labeled DICT) in the directory. The corresponding vector word set is the first vector word in the table. The table position of the first vector code word in all other vector code sets is defined relative to the space character vector word. The directory's memory word 30001 (address 00005) defines the exclamation point (!) as having three vector code words of which the first is the first word following the space character code word in the table. Note that the directory is arranged to match sequential byte codes (040 through 137); if the directory is modified, the character entries must correspond to sequential byte codes.

Additional character entries (and corresponding vector code sets) can be inserted into the directory (and the table) ahead of the space character. The added characters would have to correspond to some or all of the byte codes 000 through 037 and the BASE value would have to be changed as appropriate for the lowest byte code used. Also, the following changes would be required:

- A move of the label DICT to the statement that defines the number of vector code words for the character corresponding to the lowest byte code used.
- A redefinition of "." by an EQU .XXX, where XXX is the identification used for the first character in the table.
- Additional directory entries and table vector word sets for the added characters.

Characters corresponding to byte codes 140 through 177 octal can be added to the module by making character entries after the last one currently in the directory and by adding corresponding vector sets. If necessary, the MASK value can be increased above 177 (to a maximum of 377 octal) to accommodate more characters.

A character currently in the module can be replaced by a new character as long as:

- The new character corresponds to the byte code of the old character.
- The number of vector code words for the new character is correctly specified in the new entry in the directory.
- The vector code set for the new character replaces the old set.

Note: Some of the vector code sets in the table are used in combinations for some characters (e.g., B,P, and R). If any one of the combination characters is replaced, it may be necessary to redefine the remaining character(s).

```
 PAGE 0002 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A-91200-16002-2


0001                 ASMB,L,C
0003*
0004*      NAME:   DICTIONARY/CODE TABLE OF VIDEO DISPLAY LIBRARY.
0005*      SOURCE: 91200-18002 (FILE 4)
0006*      BINARY: 91200-16002 (MODULE 4)
0007*      PGMR:   R.M.C. B REV MADE BY DENTON ANDERSON
0008*
0009*
0010  00000                  NAM [TABL,7 91200-16002 REV.B 751222

0012                         ENT DI[T,[TAB


0014*          DIRECTORY FORMAT: .XXX-.+NN0000B

0016*          WHERE, ".XXX" IS THE NAME OF THE FIRST
0017*                 VECTOR CODE WORD;

0019*                  "NN" IS THE NUMBER OF VECTOR CODE
0020*                  WORDS FOR ".XXX".  (4 BITS).


0022************  DO NOT REARRANGE THE ORDERING OF THIS TABLE!  *******
0023  00000 000001R DI[T   DEF MASK       ADDRESS DEFINITION FOR CHAR.    *
0024  00001 000177  MASK   OCT 177        ALLOW DEFINITION OF CODE LEVEL  *
0025  00002 177740  BASE   OCT -40        NEG OF FIRST ALLOWED CODE       *
0026  00003 177700  LENTH  ABS STBL-ETbL -(# OF ENTRIES IN DICTIONARY)   *
0027*                                                                     *
0028  00004 010000  DICT   ABS .SPC-.+0100000B SPACE                      *
0029  00004         STBL   EQU DICT                                       *
0030  00005 030001         ABS .EXP-.+0300000B !                          *
0031  00006 040004         ABS .QTE-.+0400000B "                          *
0032  00007 050010         ABS .NUM-.+0500000B #                          *
0033  00010 110301         ABS .S-.+1100000B $                            *
0034  00011 100063         ABS .PCT-.+1000000B %                          *
0035  00012 060015         ABS .AND-.+0600000B &                          *
0036  00013 030023         ABS .PRM-.+0300000B '       APPOSTROPHE        *
0037  00014 030026         ABS .LP-.+0300000B (                           *
0038  00015 030031         ABS .RP-.+0300000B )                           *
0039  00016 040034         ABS .STR-.+0400000B *                          *
0040  00017 030040         ABS .PLS-.+0300000B +                          *
0041  00020 040047         ABS .CMA-.+0400000B ,       COMMA              *
0042  00021 020053         ABS .MIN-.+0200000B -                          *
0043  00022 030055         ABS .PER-.+0300000B .                          *
0044  00023 020071         ABS .SLH-.+0200000B /                          *
0045  00024 100071         ABS .0-.+1000000B 0                            *
0046  00025 030101         ABS .1-.+0300000B 1                            *
0047  00026 060104         ABS .2-.+0600000B 2                            *
0048  00027 070112         ABS .3-.+0700000B 3                            *
0049  00030 030121         ABS .4-.+0300000B 4                            *
0050  00031 060124         ABS .5-.+0600000B 5                            *
0051  00032 060132         ABS .6-.+0600000B 6                            *
0052  00033 030140         ABS .7-.+0300000B 7                            *
0053  00034 110143         ABS .8-.+1100000B 8                            *
0054  00035 060154         ABS .9-.+0600000B 9                            *
```

Figure A-3. Module [TABL (Sheet 1 of 11)

```
 PAGE 0003 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A-91200-16002-2


0055  00036 060055          ABS .CLN=.+0600000B :                          *
0056  00037 100043          ABS .SCN=.+1000000B ;                          *
0057  00040 030162          ABS .LTN=.+0300000B <                          *
0058  00041 030165          ABS .EQU=.+0300000B =                          *
0059  00042 020170          ABS .GTN=.+0200000B >                          *
0060  00043 050172          ABS .QM=.+0500000B ?                           *
0061  00044 070177          ABS .ATS=.+0700000B @                          *
0062  00045 040206          ABS .A=.+0400000B A                            *
0063  00046 070262          ABS .B=.+0700000B B                            *
0064  00047 060212          ABS .C=.+0600000B C                            *
0065  00050 040223          ABS .D=.+0400000B D                            *
0066  00051 040227          ABS .E=.+0400000B E                            *
0067  00052 030230          ABS .F=.+0300000B F                            *
0068  00053 110212          ABS .G=.+1100000B G                            *
0069  00054 030233          ABS .H=.+0300000B H                            *
0070  00055 040236          ABS .I=.+0400000B I                            *
0071  00056 050242          ABS .J=.+0500000B J                            *
0072  00057 030247          ABS .K=.+0300000B K                            *
0073  00060 020252          ABS .L=.+0200000B L                            *
0074  00061 030254          ABS .M=.+0300000B M                            *
0075  00062 030257          ABS .N=.+0300000B N                            *
0076  00063 050074          ABS .O=.+0500000B O                            *
0077  00064 040265          ABS .P=.+0400000B P                            *
0078  00065 060273          ABS .Q=.+0600000B Q                            *
0079  00066 060265          ABS .R=.+0600000B R                            *
0080  00067 070301          ABS .S=.+0700000B S                            *
0081  00070 030312          ABS .T=.+0300000B T                            *
0082  00071 040315          ABS .U=.+0400000B U                            *
0083  00072 040321          ABS .V=.+0400000B V                            *
0084  00073 060325          ABS .W=.+0600000B W                            *
0085  00074 040333          ABS .X=.+0400000B X                            *
0086  00075 050337          ABS .Y=.+0500000B Y                            *
0087  00076 040344          ABS .Z=.+0400000B Z                            *
0088  00077 030350          ABS .LBK=.+0300000B [                          *
0089  00100 020353          ABS .RSL=.+0200000B \                          *
0090  00101 030355          ABS .RBK=.+0300000B ]                          *
0091  00102 030360          ABS .UAW=.+0300000B ↑                          *
0092  00103 040363          ABS .LAW=.+0400000B ←                          *
0093  00104           ETBL  EQU *                                          *
0094************  DO NOT REARRANGE THE CONTENTS OF THIS TABLE!  *******
0095*          END OF "ASCII CODE DIRECTORY".
```

Figure A-3. Module [TABL (Sheet 2 of 11)

```
 PAGE 0004 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A-91200-16002-2


0097*          VECTOR BYTE CODES ARE PACKED TWO(2) PER
0098*          VECTOR CODE WORD.

0100*          FORM: CCDDDLLL,CCDDDLLL

0102*          WHERE, C/CODE: 0 = NON-WRITING,
0103*                         1 = FIRST AND LAST POINTS
0104*                             ARE SUPPRESSED (IT IS
0105*                             USED TO DRAW A VECTOR
0106*                             BETWEEN TWO(2) OTHER
0107*                             VECTORS),
0108*                         2 = FULL VECTOR (ALL POINTS
0109*                             ARE DISPLAYED),
0110*                         3 = FIRST POINT IS
0111*                             SUPPRESSED (IT IS USED
0112*                             TO CONTINUE FROM THE
0113*                             END OF AN EXISTING
0114*                             VECTOR);

0116*                  D/DIRECTION CODES: STARTING WITH
0117*                          ZERO(0) DEGREES ON THE UNIT
0118*                          CIRCLE, THERE ARE EIGHT (8)
0119*                          POSSIBLE DIRECTIONS AT : 0,45,90,135,180,
0120*                          225,270,AND 315 DEGREES RESPECTIVELY.
0121*                          RANGE = 0:7

0123*                  L/LENGTH: MAY BE 0 TO 7 UNITS LONG.


0125*       WATCH OUT FOR COMBINED CODES FOR THE FOLLOWING GROUPS:-

0127*       [B P R]   [C G]   [E F]   [% / 0 O]   [, ;]   [. :]   [$ S]


0129  00104 000105R [TAB  DEF .          DEFINES THE CODE TABLE BASE ADDR&.
0130  00105 003000  .SPC  OCT 3000     (SPACE).
0131  00105           .    EQU .SPC
0132  00106 001200  .EXP  OCT 1200,11224,36062 !
      00107 011224
      00110 036062
0133  00111 004423  .QTE  OCT 4423,111002,131073,30400 "
      00112 111002
      00113 131073
      00114 030400
0134  00115 011204  .NUM  OCT 11204,11244,4664,1224,35462 #
      00116 011244
      00117 004664
      00120 001224
      00121 035462
0135  00122 002234  .AND  OCT 2234,150711,174761,165361,174701,145072 &
      00123 150711
      00124 174761
      00125 165361
      00126 174701
      00127 145072
```

Figure A-3. Module [TABL (Sheet 3 of 11)

```
 PAGE 0005 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A-91200-16002-2


0136  00130 005022  .PRM  OCT 5022,111074,31000 !
      00131 111074
      00132 031000
0137  00133 005423  .LP   OCT 5423,124764,174403 (
      00134 124764
      00135 174403
0138  00136 012411  .RP   OCT 12411,134764,164405 )
      00137 134764
      00140 164405
0139  00141 010614  .STR  OCT 10614,21264,1234,36401    *
      00142 021264
      00143 001234
      00144 036401
0140  00145 011604  .PLS  OCT 11604,15264,34403 +
      00146 015264
      00147 034403
0141  00150 005022  .SCN  OCT 5022,110741,170501,31052  ;  (USES COMMA)
      00151 110741
      00152 170501
      00153 031052
0142  00154 034611  .CMA  OCT 34611,151341,170501,34403    ,
      00155 151341
      00156 170501
      00157 034403
0143  00160 011011  .MIN  OCT 11011,101073 -
      00161 101073
0144  00162 000601  .PER  OCT 601,150741,70405 .
      00163 150741
      00164 070405
0145  00165 016241        OCT 16241,170701,50474    2ND DOT FOR :
      00166 170701
      00167 050474
0146  00170 012221  .PCT  OCT 12221,140761,60473,110701,170541,21461  %
      00171 140761
      00172 060473
      00173 110701
      00174 170541
      00175 021461
0147  00176 010614  .SLH  OCT 10614,31472 /
      00177 031472
0148  00176        .0     EQU .SLH
0149  00200 000046        OCT 46    0    (USES / & 0)
0150  00201 000631  0X    OCT 631,152311,141371,172351,61005   0
      00202 152311
      00203 141371
      00204 172351
      00205 061005
0151  00206 000602  .1     OCT 602,20726,164475 1
      00207 020726
      00210 164475
0152  00211 012611  .2     OCT 12611,141371,170751,161351,171304,1000 2
      00212 141371
      00213 170751
      00214 161351
      00215 171304
      00216 001000
```

Figure A-3. Module [TABL (Sheet 4 of 11)

```
 PAGE 0006 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A=91200-16002-2


0153   00217 012611   .3     OCT 12611,141371,170751,160452,134702,144721 3
       00220 141371
       00221 170751
       00222 160452
       00223 134702
       00224 144721
0154   00225 054473          OCT 54473    3
0155   00226 001626   .4     OCT 1626,165761,142072 4
       00227 165761
       00230 142072
0156   00231 010671   .5     OCT 10671,141311,151331,161722,142072,64 5
       00232 141311
       00233 151331
       00234 161722
       00235 142072
       00236 000064
0157   00237 000631   .6     OCT 631,151712,140453,141771,170751,61005 6
       00240 151712
       00241 140453
       00242 141771
       00243 170751
       00244 061005
0158   00245 013204   .7     OCT 13204,170753,171005 7
       00246 170753
       00247 171005
0159   00250 005611   .8     OCT 5611,150731,161351,170771,141371,170751 8
       00251 150731
       00252 161351
       00253 170771
       00254 141371
       00255 170751
0160   00256 161331          OCT 161331,150711,1073 8
       00257 150711
       00260 001073
0161   00261 101312   .9     OCT 101312,151731,161351,170771,41472,61     9
       00262 151731
       00263 161351
       00264 170771
       00265 041472
       00266 000061
0162   00162          .CLN   EQU .PER      :
0163   00267 005422   .LTN   OCT 5422,125372,34402 <
       00270 125372
       00271 034402
0164   00272 005642   .EQU   OCT 5642,30602,35001 =
       00273 030602
       00274 035001
0165   00275 004612   .GTN   OCT 4612,155075 >
       00276 155075
0166   00277 012611   .QM    OCT 12611,140771,170751,170462,100004 ?
       00300 140771
       00301 170751
       00302 170462
       00303 100004
```

Figure A-3. Module [TABL (Sheet 5 of 11)

```
 PAGE 0007 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A-91200-16002-2


0168   00304 012611   .ATS  OCT 12611,141371,172351,161331,150711,140563 @
       00305 141371
       00306 172351
       00307 161331
       00310 150711
       00311 140563
0169   00312 001002         OCT 1002     @
0170   00313 112312   .A    OCT 112312,175364,15042,42072 A
       00314 175364
       00315 015042
       00316 042072
0171*                  .B    LOCATED BEFORE .P
0172   00317 010624   .C    OCT 10624,144702,174454,174702,144471,400 C
       00320 144702
       00321 174454
       00322 174702
       00323 144471
       00324 000400
0173   00325 020433   GX    OCT 20433,101162,34401      G
       00326 101162
       00327 034401
0174   00330 113303   .D    OCT 113303,174764,164543,3000 D
       00331 174764
       00332 164543
       00333 003000
0175   00334 142044   .E    OCT 142044    E
0176   00335 113304   .F    OCT 113304,25441,141473 F
       00336 025441
       00337 141473
0177   00317          .G    EQU .C
0178   00340 113063   .H    OCT 113063,42023,133002 H
       00341 042023
       00342 133002
0179   00343 000602   .I    OCT 602,20526,20602,35463 I
       00344 020526
       00345 020602
       00346 035463
0180   00347 010671   .J    OCT 10671,140711,52441,101072,64 J
       00350 140711
       00351 052441
       00352 101072
       00353 000064
0181   00354 113064   .K    OCT 113064,146053,175402 K
       00355 146053
       00356 175402
0182   00357 013266   .L    OCT 13266,142002 L
       00360 142002
0183   00361 113372   .M    OCT 113372,145366,1000 M
       00362 145366
       00363 001000
0184   00364 113061   .N    OCT 113061,76025,133002 N
       00365 076025
       00366 133002
0185   00201          .O    EQU 0X
```

Figure A-3. Module [TABL (Sheet 6 of 11)

```
 PAGE 0008 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A-91200-16002-2


0187  00367 141711  .B    OCT 141711,150531,25400     B  (USES P)
      00370 150531
      00371 025400
0188  00372 113303  .P    OCT 113303,174761,164543,35403 P
      00373 174761
      00374 164543
      00375 035403
0189  00376 015442  RX    OCT 15442,175402   R
      00377 175402
0190  00400 001241  .Q    OCT 1241,154724,144702,174763,65022,135002   Q
      00401 154724
      00402 144702
      00403 174763
      00404 065022
      00405 135002
0191  00372         .R    EQU .P
0192  00406 010671  .S    OCT 10671,141311,150731,161331,150711,141371
      00407 141311
      00410 150731
      00411 161331
      00412 150711
      00413 141371
0193  00414 035063        OCT 35063    S
0194  00415 022126        OCT 22126,36062     $
      00416 036062
0195  00417 013204  .T    OCT 13204,21366,2000 T
      00420 021366
      00421 002000
0196  00422 013265  .U    OCT 13265,174702,144725,35064 U
      00423 174702
      00424 144725
      00425 035064
0197  00426 013264  .V    OCT 13264,175312,152072,32000 V
      00427 175312
      00430 152072
      00431 032000
0198  00432 013265  .W    OCT 13265,174711,151062,174711,152472,32000 W
      00433 174711
      00434 151062
      00435 174711
      00436 152472
      00437 032000
0199  00440 110714  .X    OCT 110714,150444,130774,170402 X
      00441 150444
      00442 130774
      00443 170402
0200  00444 013261  .Y    OCT 13261,175312,150452,30763,2000 Y
      00445 175312
      00446 150452
      00447 030763
      00450 002000
0201  00451 013204  .Z    OCT 13204,170754,170704,1000 Z9
      00452 170754
      00453 170704
      00454 001000
```

Figure A-3. Module [TABL (Sheet 7 of 11)

```
 PAGE 0009 #01  VIDEO SUBROUTINE [TABL (ASCII CODE TABLE)A-91200-16002-2


0203   00455 011413   .LBK   OCT 11413,121366,141002 {
       00456 121366
       00457 141002
0204   00460 012674   .RSL   OCT 12674,34401 \
       00461 034401
0205   00462 012411   .RBK   OCT 12411,101366,161004 }
       00463 101366
       00464 161004
0206   00465 012212   .UAW   OCT 12212,175032,173004 ↑
       00466 175032
       00467 173004
0207   00470 011612   .LAW   OCT 11612,25372,15304,35061    ←  *
       00471 025372
       00472 015304
       00473 035061

0209                          END
**  NO ERRORS* RTE ASMB 92001B (10/74)**
```

Figure A-3. Module [TABL (Sheet 8 of 11)

```
PAGE 0010                    [TABL   91200-16002 REV.b 751222

                    CROSS-REFERENCE SYMBOL TABLE


.       00131   00028   00030   00031   00032   00033   00034
        00035   00036   00037   00038   00039   00040   00041
        00042   00043   00044   00045   00046   00047   00048
        00049   00050   00051   00052   00053   00054   00055
        00056   00057   00058   00059   00060   00061   00062
        00063   00064   00065   00066   00067   00068   00069
        00070   00071   00072   00073   00074   00075   00076
        00077   00078   00079   00080   00081   00082   00083
        00084   00085   00086   00087   00088   00089   00090
        00091   00092   00129

.0      00148   00045

.1      00151   00046

.2      00152   00047

.3      00153   00048

.4      00155   00049

.5      00156   00050

.6      00157   00051

.7      00158   00052

.8      00159   00053

.9      00161   00054

.A      00170   00062

.AND    00135   00035

.ATS    00168   00061

.B      00187   00063

.C      00172   00064   00177

.CLN    00162   00055

.CMA    00142   00041

.D      00174   00065

.E      00175   00066

.EQU    00164   00058

.EXP    00132   00030

.F      00176   00067
```

Figure A-3. Module [TABL (Sheet 9 of 11)

```
PAGE 0011                          [TABL   91200-16002 REV.B 751222

                    CROSS-REFERENCE SYMBOL TABLE

.G      00177    00068
.GTN    00165    00059
.H      00178    00069
.I      00179    00070
.J      00180    00071
.K      00181    00072
.L      00182    00073
.LAW    00207    00092
.LBK    00203    00088
.LP     00137    00037
.LTN    00163    00057
.M      00183    00074
.MIN    00143    00042
.N      00184    00075
.NUM    00134    00032
.O      00185    00076
.P      00188    00077    00191
.PCT    00146    00034
.PER    00144    00043    00162
.PLS    00140    00040
.PRM    00136    00036
.Q      00190    00078
.QM     00166    00060
.QTE    00133    00031
.R      00191    00079
.RBK    00205    00090
.RP     00138    00038
```

Figure A-3. Module [TABL (Sheet 10 of 11)

```
PAGE 0012                          [TABL   91200-16002 REV.B 751222

                    CROSS-REFERENCE SYMBOL TABLE

 .RSL    00204    00089
 .S      00192    00033    00080
 .SCN    00141    00056
 .SLH    00147    00044    00148
 .SPC    00130    00028    00131
 .STR    00139    00039
 .T      00195    00081
 .U      00196    00082
 .UAW    00206    00091
 .V      00197    00083
 .W      00198    00084
 .X      00199    00085
 .Y      00200    00086
 .Z      00201    00087
#BASE    00025
 DICT    00028    00029
 DI[T    00023    00012
 ETBL    00093    00026
#GX      00173
#LENTH   00026
 MASK    00024    00023
 OX      00150    00185
#RX      00189
 STBL    00029    00026
 [TAB    00129    00012
```

Figure A-3. Module [TABL (Sheet 11 of 11)

ASSEMBLY LANGUAGE I/O INSTRUCTIONS

APPENDIX B

This appendix describes the results of Assembly Language I/O instructions addressed to the TV interface card. Table B-1 describes the TV card's response to Assembly I/O instructions and Figure B-1 is an example of Assembly Language I/O programming

Note: This appendix is provided for informational purposes only. The use of Assembly Language I/O instructions in a user-written RTE program (other than an I/O driver) will cause a Memory Protect Violation to occur and, consequently, the program will be aborted.

Table B-1. Assembly Language I/O Instructions for TV Interface Card

INSTRUCTION	DESCRIPTION
CLC xx	Clears interrupt control flip-flop. Sets mode flip-flop such that next output to card must be a mode word.
CLC 0 (See Note.)	(CRS). Same as CLC xx plus it resets all counters.
POPIO (I/O signal*)	Sets flag flip-flop, requests complete screen (bulk) erase, and presets for normal video polarity.
STF xx	Sets flag flip-flop.
SFC xx	Causes program skip if flag is clear.
SFS xx	Causes program skip if flag is set.
STC xx	Sets the interrupt control flip-flop and (following point address OTA/B's) sets the write request flip-flop which will cause the addressed point to be written or erased depending on the mode.
CLF xx	Clears the flag flip-flop and interlock flip-flop to allow next interrupt. This is generally issued as STC xx,C or OTA xx,C.
LIA xx or LIB xx or MIA xx or MIB xx	Any input request will set the bulk-erase request flip-flop. It should be followed by a CLF or STC xx,C to allow flag check or interrupt at the end of the bulk erase operation.
OTA xx or OTB xx	If following CLC (mode flip-flop set), bits 0 and 1 will be strobed into mode storage register as shown in tables below. (CLF must be issued after or with OTA/B to clear the mode flip-flop.) Following the mode OTA/B and CLF, each OTA/OTB strobes all 16 bits into the point address input registers.

Note: A CLC 0 instruction should not be used in any operating system environment (RTE, BCS, DOS, etc.) that has established the desired operating conditions of the system's I/O devices. CLC 0 clears the control flip-flop on all I/O interface cards in the system. Consequently, ongoing data transfers might be prematurely terminated and attention-demanding interrupts might be prevented.

*POPIO is an I/O signal (not an instruction) which, along with CRS, is generated when the computer's PRESET (or EXTERNAL PRESET) switch is pressed.

Table B-1. Assembly Language I/O Instructions for TV Interface Card (Continued)

INSTRUCTION

OTA xx
or OTB xx

DESCRIPTION

COLOR	CONWD MODE CODE	CONWD BITS			OTA MODE WORD BITS			MODE STORAGE REGISTER BIT 0		
		10	9	6	3	2	0	CARD A (RED)	CARD B (GREEN)	CARD C (BLUE)
White	0	0	0	0	0	0	0	0	0	0
Black	1	0	0	1	1	1	1	1	1	1
Red	2	0	1	0	1	1	0	0	1	1
Green	3	0	1	1	1	0	1	1	0	1
Blue	4	1	0	0	0	1	1	1	1	0
Yellow	5	1	0	1	1	0	0	0	0	1
Magenta	6	1	1	0	0	1	0	0	1	0
Cyan	7	1	1	1	0	0	1	1	0	0

VIDEO POLARITY	CONMD BIT 7	OTA MODE WORD BIT 0	MODE STORAGE REGISTER BIT 1		
			CARD A (RED)	CARD B (GREEN)	CARD C (BLUE)
Normal	0	0	0	0	0
Inverse	1	1	1	1	1

```
          .
          .
          .
          LIA XX       REQUEST BULK ERASE.
          STC,XX,C     CLF WOULD DO UNLESS INTERRUPT NEEDED.
          LDA BUFAD    SET UP POINTER TO
          STA PNTR        BUFFER ADDRESS.
          LDB NUMBR    SET POINT
          CMB,INB         COUNTER.
          LDA MODE     GET MODE WORD FOR OUTPUT.
          SFS XX       ERASE DONE?
          JMP *-1      NO.
          CLC XX       YES, SET FOR MODE.
          OTA XX       OUTPUT MODE WORD.
          STC,XX,C     CLF WOULD DO UNLESS INTERRUPT NEEDED.
MORE      LDA PNTR,I   GET A POINT.
          SFS XX       PREVIOUS OUTPUT DONE?
          JMP *-1      NO.
          OTA XX       YES, OUTPUT NEXT POINT.
          STC XX,C     MUST HAVE STC HERE TO DISPLAY THE POINT.
          ISZ PNTR     INCREMENT BUFFER ADDRESS.
          INB,SZB      ALL DONE?
          JMP MORE     NO, GO DO NEXT POINT.
          (CONTINUE)
          .
          .
          .
          .
BUFAD     DEF BUF      ADDRESS OF POINT BUFFER
NUMBR     DEC N        NUMBER OF POINTS (N)
PNTR      BSS 1        CURRENT POINT BUFFER ADDRESS
BUF       BSS N        POINT BUFFER (N = NUMBER OF POINTS)
MODE      OCT M        REFER TO TABLE B-1 FOR VALUE OF M
```

Figure B-1. Example of Assembly Language I/O Programming

HEWLETT-PACKARD COMPANY
11000 WOLFE ROAD, CUPERTINO, CALIFORNIA, 95014

PART NO. 91200-90006

Printed in U.S.A. APR 1977