Chapter 20: Obsolete Codes

Contents of this Chapter

This chapter discusses the following PCL commands:

20.1 Procedure for Obsoleting Functionality

The procedure for obsoleting a functionality is:

1. Move the obsolete functionality to the obsolete section of the the Implementor's Guide.

2. Products that contain the obsolete functionality must stop documenting the functionality.

3. After two product generations, when most software supporting the product has stopped utilizing the functionality, it can be removed from the device.

20.2 Movement Segment

Clear All Horizontal Tabs   *Esc 3*

Clears all horizontal tab stops.

Clear Horizontal Tab at Current Column   *Esc 2*

Clears the horizontal tab stop at CAP, if one is set.

Horizontal Tabulation   *Esc 1*

Prforms the same function as the Horizontal Tab (HT) character.

Set Horizontal Tab   *Esc & a # t/T*

Sets a horizontal tab stop at the specified column.

Value(#)   =   Column number
Range =   0 to printer's physical limits (unspecified values ignored)

Set Horizontal Tab at Current Column   *Esc 1*

Sets a horizontal tab stop at CAP.

Clear Horizontal Tab   *Esc & a # u/U*

Clears the horizontal tab stop at the specified column, if one is set.

Value(#)   =   Column number
Range   =   0 to printer's physical limits (unspecified values ignored)

Vertical Tab   *VT*

Moves CAP to the same horizontal position on the next vertical tab stop. If no such line exists, VT moves CAP to the first line of the next logical page (i.e., to the default vertical tab stop at top of form).

Half Line Feed   *Esc =*

Moves the current active position to the same horizontal position 1/2 line down.

Set Vertical Tab (Relative)   *Esc & l 1 m/M*

Sets a vertical tab stop at the current line. It should be possible to set a vertical tab stop in every line within the logical page.

Clear Vertical Tab (Relative)   *Esc & l 2 m/M*

Clears the vertical tab stop at the current line, if one is set. The tab stop at top of form cannot be cleared.

Clear All Vertical Tabs   *Esc & l 3 m/M*

Clears all vertical tab stops, except the tab at top of form.

Clear Vertical Tab (Absolute)   *Esc & l # r/R*

Clears the vertical tab stop in the specified line, if one is set.

```
Value(#)   =   Line number
Default    =   One vertical tab at top of form
Range      =   0  to  logical page bound (unspecified values ignored)
```

The vertical tab stop at top of form cannot be cleared.

Set Vertical Tab (Absolute)   *Esc & l # y/Y*

Sets a vertical tab in the specified line specified.

```
Value(#)   =   Line number
Default    =   One vertical tab at top of form
Range      =   0  to  printer's physical limits (unspecified values ignored)
```

The first line within a page is line 0 at the top of form.

Print with Paper Motion via VFC   *Esc & l # v/V*

Moves CAP to the first character position of the next line referenced by the specified VFC channel.

Value(#)   =   Channel number
Default   =   NA
Range   =   0 to 16

If no line is referenced, CAP moves to the left margin at the top of form of the next page.

A printer should support a 16 channel VFC. If a nonexistent channel (e.g., 17) is referenced, this command causes a CR-LF. A value field of 0 is a special case, causing the printer to move paper to the left margin and top of form of the next page, unless already there.

Define VFC Table   *Esc & l # W[VFC data]*

Programs the VFC with the data bytes immediately following the terminating character.

Value(#)   =   Number of data bytes following the terminating character
Default   =   NA
Range   =   0 to 256

The data field consists of binary-packed two-byte words, with the most significant byte sent first. Each two-byte word represents one line. The most significant bit of the most significant byte describes channel 16; the least significant bit of the least significant byte describes channel 1. The maximum byte count allowed is twice the maximum supported physical page length. If (byte count / 2) is less than page length, the rest of the VFC table is filled with zeros. If (byte count / 2) is greater than page length, or if the line spacing is 0, or when the byte count is odd, the downloaded VFC is discarded. A 1 in a bit position references that line.

The first occurrence of channel 2 determines text length: if channel 2 is clear, or the first stop is outside the end of the page, text length is equal to the end of logical page. If an orientation, page length, top margin, text length, or line spacing command is received, the programmed VFC is removed and a default VFC calculated from the current line spacing, page length, top margin and text length settings. If the requested VFC table has more lines than the maximum supported, the VFC table is truncated at its maximum limit. If a value field of 0 is received, a default VFC is calculated from the default top margin and text length.

If an illegal command is received, the associated data is stripped so it is not misinterpreted (i.e. neither printed nor acted upon).

NOTE: It is recommended that the VFC when defaulted should not be recalculated until needed (i.e., receipt of *Esc & l # V*); calculating a default VFC table is time consuming.

Default VFC

The default definition for each of the 16 VFC channels is as follows:

Channel	Definition
1	Top of form
2	Bottom of form
3	Single spacing
4	Double spacing
5	Triple spacing
6	Half form
7	Quarter form
8	Tenth line
9	Bottom of form
10	Bottom of form - 1
11	Top of form - 1
12	Top of form
13	Seventh line
14	Sixth line
15	Fifth line
16	Fourth line

NOTE: VFCs, though not required, may be needed for some EDP system level printers.

Definitions

The number of a line is LINE_NUMBER. The first line of a form (i.e. specified by top margin) is line zero. The last line of a page is END_OF_PAGE. The last line of a form is END_OF_FORM. The operator *div* indicates integer division. The operator *leq* indicates "less than or equal".

Channel 1 is active at line 0 only.

Channel 2 is active at END_OF_FORM only.

Channel 3 is active for all lines from 0 through END_OF_FORM.

Channel 4 is active for all lines where LINE_NUMBER is evenly divisible by 2 and 0 *leq* LINE_NUMBER *leq* END_OF_FORM.

Channel 5 is active for all lines where LINE_NUMBER is evenly divisible by 3 and 0 *leq* LINE_NUMBER *leq* END_OF_FORM.

Channel 6 is active for lines 0 and (END_OF_FORM + 2) *div 2*.

Channel 7 is active for lines 0 , (END_OF_FORM + 4) *div 4*, (END_OF_FORM + 2) *div 2*, and (3 * END_OF_FORM + 6) *div 4*.

Channel 8 is active for all lines where LINE_NUMBER is evenly divisible by 10 and 0 *leq* LINE_NUMBER *leq* END_OF_FORM.

Channel 9 is active at END_OF_FORM only.

Channel 10 is active at END_OF_FORM - 1 only.

Channel 11 is active at END_OF_PAGE only.

Channel 12 is active at line 0 only.

Channel 13 is active for all lines where LINE_NUMBER is evenly divisible by 7 and 0 *leq* LINE_NUMBER *leq* END_OF_FORM.

Channel 14 is active for all lines where LINE_NUMBER is evenly divisible by 6 and 0 *leq* LINE_NUMBER *leq* END_OF_FORM.

Channel 15 is active for all lines where LINE_NUMBER is evenly divisible by 5 and 0 *leq* LINE_NUMBER *leq* END_OF_FORM.

Channel 16 is active for all lines where LINE_NUMBER is evenly divisible by 4 and 0 *leq* LINE_NUMBER *leq* END_OF_FORM.

Example

The following is an example default VFC for an 11-inch page at 6 lpi (line zero corresponds to the top margin setting). Page length is 66 lines (END_OF_PAGE = 65) and the text length is 60 lines (END_OF_FORM = 59). VFC lines are relative to the top margin, which defaults to 1/2 inch from the top of the physical page (depending on the perforation skip).

Channel	Lines Active
1	0
2	59
3	0, 1, 2, ...59
4	0, 2, 4, ...58
5	0, 3, 6, ...57
6	0, 30
7	0, 15, 30, 45
8	0, 10, 20, ...50
9	59
10	58
11	65
12	0
13	0, 7, 14, ...56
14	0, 6, 12, ...54
15	0, 5, 10, ...55
16	0, 4, 8, ...56

20.3 Rendering Segment

Horizontal Raster Resolution   *Esc * r # l/L*
Vertical Raster Resolution   *Esc * r # v/V*

These commands allow horizontal and vertical raster graphics resolutions to be set independently.

Value(#)   =   Resolution in dots per inch (dpi)
Default   =   Device dependent
Range   =   0 to 32767

GPIS Data Binding (GPIS Vector Graphics)   *Esc * t # g/G*

Specifies the vector graphics binding of the data in the data transfer escape sequence (i.e., *Esc*t#W*). The default binding is "none"; a binding must be specified on at least the first transfer of vector graphics data. In the event that vector graphics data is received with an unsupported or no binding specified, the printer will strip and discard the data.

Binding #	Binding Type
0	Exit all vector graphics modes (Default)
1	16-bit binary GPIS binding
2	8-bit binary GPIS binding
3	Character-encoded GPIS binding

Horizontal Picture Dimension   *Esc * t # h/H*

NOTE: This sequence has been reassigned to Destination Raster Width (Chapter 13).

The value field specifies the width of the vector graphics picture in decipoints. This value is only meaningful if the device is in Printer Mode. The picture dimension has precedence over the scaling information embedded in subsequent vector graphics metafiles, effectively making all metafile data of the picture library type. The default horizontal size in Printer Mode is the width of the logical page. If either the horizontal or vertical dimensions are 0, the printer does not override the scaling information included in the subsequent vector graphics metafile.

Vector Graphics Operating Mode   *Esc * t # m/M*

This escape sequence specifies the operating mode of the printer. A 0 in the value field, the default mode, specifies Plotter Mode. In Plotter Mode all vector graphics data is interpreted literally (i.e., the presentation directives and the print control comes from commands within the vector graphics data stream). A 1 in the value field specifies Printer Mode. In Printer Mode escape sequences control the presentation and printing of the image, overriding the information embedded in the vector graphics data stream. This mode allows the user to control the presentation and printing of the picture without altering the vector graphics data.

Vector Graphics Mapping Mode   *Esc * t # n/N*

Mapping mode specifies whether distortion should be permitted when mapping the picture image to the user-specified height and width. This value is only meaningful if the device is in Printer Mode. The default Mapping Mode in Printer Mode is 1 (undistorted). A value field of 2 specifies that distortion is allowed, and a value of 0 defaults to the vector graphics file mapping mode.

Print Control   *Esc * t # p/P*

This command controls the printing of vector graphics images. A value field of 0 specifies that the image should be printed and retained. A value field of 1 specifies that the vector graphics picture should be printed and cleared. The picture will be printed with the upper left hand corer of the image at the current active position.

Print Mode (Graphics)   *Esc * p # n/N*

Defines the print mode or printing-motion algorithm for graphics.

```
Value(#)   =   0   Graphics default (no-break algorithm)
           =   1   Print graphics bidirectionally
           =   2   Print graphics left to right
           =   3   Print graphics right to left
           =   4   Conditionally print bidirectional using no-break or smart
                   bidirectional algorithm *
Default    =   0
Range      =   0 to 4 (command is ignored for unsupported values)
```

* DEVICE NOTE: DJ550C prints bidirectional if there are four or more blank raster rows between print passes. DJ500 prints bidirectional if there is only a single blank raster row between passes.

Print Mode (Text)   *Esc & k # w/W*

Defines the print mode or printing-motion algorithm for text. See the individual product guides for the meaning of specific values.

Value(#)   =   0   Unidirectional (left-to-right)
           =   1   Bidirectional mode
           =   2   Unidirectional (right-to-left)
           =   3   High intensity mode
           =   5   Text scale OFF
           =   6   Text scale ON
           =   7   Fast high intensity
Default   =   Device dependent (recommend bidirectional)
Range   =   Device dependent

"High intensity" is used for special printer technology media combinations (e.g., thermal inkjet on transparency).

"Text scale ON" allows 66 lines at 6 lpi, or 88 lines at 8 lpi to print on an 11 inch page, even if the effective paper length is shorter than 11 inches. Thus, on an 11 inch page with an 1/2 inch unprintable region, 66 lines (at 6 lpi) would still be printed.

DEVICE NOTE: PJ uses values of 0, 1, 3.

DEVICE NOTE: DJ uses values of 0, 1, 2, 5, 6.

NOTE: This escape sequence terminated with a *W* is an exception in that it **does not** include binary data.

Raster Graphics Quality   *Esc * r # q/Q*

Determines the quality with which graphics is printed.

Value(#)   =   0   User-selected
           =   1   Draft
           =   2   High quality
Default   =   0
Range   =   0 to 2

Selecting a value of 1 causes raster graphics to be printed at nearly double the normal speed. Print quality is traded for ink savings and increased throughput. The graphics quality is not reset by the End Raster Graphics command (*Esc*rC*).

Vertical Picture Dimension   *Esc * t # v/V*

NOTE: This sequence has been reassigned to Destination Raster Height (Chapter 13).

The value field specifies the length of the vector graphics picture in decipoints. This value is only meaningful if the device is in Printer Mode. The picture dimension has precedence over the scaling information embedded in subsequent vector graphics metafiles. The default vertical size in Printer Mode is the current text length of the logical page. If either the horizontal or vertical dimensions ar 0, the printer does not override the scaling information included in the subsequent vector graphics metafile.

GPIS Data Transfer   *Esc * t # W [GPIS Data]*

There are two methods of sending data to a vector graphics device: by putting it into a mode where it interprets all data as vector graphics commands (referred to a "stream" mode), or by "encapsulating" the data in an escape sequence. Some devices will implement both modes, the choice of which depends on the operating environment.

Bar Code Label Placement   *Esc * z # c/C*

Moves CAP to the specified column position on the pending line.

Value(#)   =   Absolute column position (+ or - is ignored)
Default   =   NA
Range   =   NA (values less than the current active position (CAP) move the CAP one
                   column to the right)

The value is based on the pitch of the current character set, or HMI if it is implemented.

This command provides device-independent bar code positioning. Like the Horizontal Positioning (*Esc&a#C*) command, starting location is specified by a column number. Horizontal spacing is based on the current font pitch, or on HMI if it is implemented.

NOTE: Because a bar code has no standard width, the width can be device specific.

Bar Code Label Height   *Esc * z # h/H*

Defines the height of the label in tenths of an inch.

Value(#)   =   Label height (tenths of an inch)
Default   =   6 (6/10 of an inch)
Range   =   0 to 32767

A value of 0 sets bar code height to the current line spacing (e.g., 6 or 8 lines per inch).

DEVICE NOTE: The HP 2934A product supports a range of 0 to 99.

Bar Code Header Control   *Esc * z # q/Q*

Controls the printing of the bar code header string.

```
Value(#)    =    0        Disables the printing of the header
            =    1        Prints the header above the bar code
            =    2        Prints the header below the bar code
Default     =    1
Range       =    0 to 2
```

DEVICE NOTE: The HP 2934A supports a range of 0-1. Printing the header below the bar code is not supported. The HP 2934A prints all valid characters in a bar code label string and ignores illegal characters. The HP 2934A truncates at the printer's physical right limit, not the right margin.

Bar Code Label   *Esc * z <Bar code> z/Z*

Sends the bar code, specified by an alphanumeric string in angle brackets, to the printer.

When execution this command, the device prints all buffered bar code data, moves to the left margin, and spaces down the height of the bar code. Start/Stop and Check characters are added as necessary, as defined by the current bar code selected

DEVICE NOTE: If an illegal character in a bar code is received, the 256x products make the bar code unreadable. The 256x products truncate the bar code at the right margin.

Bar Code Label X Offset   *Esc * z # x/X*

Defines the relative horizontal offset in dots for the next label.

```
Value(#)    =    Relative horizontal offset (+ or - is ignored)
Default     =    NA
Range       =    0 to 32767
```

The value field specifies the number of dots the label is offset to the right.

DEVICE NOTE: For the HP 2934A product, the dot spacing is 1/90 inch.

Bar Code Selection   *Esc * z # v/V*

Selects the bar code type.

Value	Bar Code Type
0	Code 3 of 9
1	Industrial 2 of 5
2	Matrix 2 of 5
3	User-defined
4	2 of 5 interleaved
5	Codabar
6	MSI/Plessey
8	UPC A
9	UPC E
10	EAN 8
11	EAN 13
12	Code 128
13	Code 93
14	Extended code 3 of 9

Default =   0
Range   =   0 to 14 (unsupported types are ignored: the previous type remains in effect)

Illegal or unsupported bar code types are ignored; the previously selected bar code type remains in effect.

DEVICE NOTE: The HP 256X products support 0,1,4,8,9,10, and 11. The 256x products cannot merge bar codes of different types on the same line.

DEVICE NOTE: The HP 2934A supports values 0 to 4.

Bar Code Print Density:

*Esc * z # r/R*   Bar Code Wide Bar Width
*Esc * z # s/S*   Bar Code Narrow Bar Width
*Esc * z # t/T*   Bar Code Wide Space Width
*Esc * z # u/U*   Bar Code Narrow Space Width

Specify the dot width of the wide bar, narrow bar, wide space, and narrow space elements.

Value(#)   =   Width in dots (half dot increments; 0.5)
Default   =   Device dependent
Range   =   1 to 9.5

The width of these elements determine the density at which a bar code symbol is printed.

DEVICE NOTE: For the HP 2934A, the dot spacing is 1/90 inch.

DEVICE NOTE: The HP 256X products do not support this sequence.

End Raster Graphics   *Esc * r B*

This command ends Raster Mode. It signifies the end of the transfer of a raster graphics image and ends the current raster row. Commands locked out by Start Raster Graphics (*Esc*r#A*) are enabled. If a value field is received, it is ignored and the command still executed.

NOTE: It is recommended that *Esc*rC* be used rather than *Esc*rB*.

X Offset   *Esc * b # x/X*

Defines a temporary horizontal offset, which is to be in effect only for the next raster line received.

Value(#)   =   Number of pixels in the offset (should be a multiple of 8) *
Default   =   NA
Range   =   0 to 32767

* Values that are not a multiple of eight are rounded down to the nearest multiple.

The X Offset plus the left graphics margin determines the horizontal location of the next raster line.

Text Color   *Esc & v # s/S*

Selects the predefined color for text.

Value(#)   =   0   Black
           =   1   Red
           =   2   Green
           =   3   Yellow
           =   4   Blue
           =   5   Magenta
           =   6   Cyan
           =   7   White
Default   =   0
Range   =   0 to 7 (command is ignored for other values)

This command uses the above fixed color table, which is different from the color palette used for raster graphics.

Configure Raster Data (format 1)   *Esc * g # W*

The Configure Raster Data (CRD) command configures the device for complex raster data transfers.

Value(#)   =   Number of data bytes
Default   =   NA
Range   =   6 to 4,294,967,295

DEVICE NOTE: DeskJet 520, and DeskJet 560C support a maximum range of 32767.

DEVICE NOTE: DeskJet 850C locks out Simple Color (*Esc*r#U*) and Raster Resolution (*Esc*t#R*) after a valid CRD until the receipt of an *EscE* or equivalent device reset, or a CRD command with a value field of 0 (*Esc*g0W*). A CRD command with a 0 value field unlocks Simple Raster (*Esc*r#U*) and Raster Resolution (*Esc*t#R*), resets the horizontal and vertical resolutions to 75 ppi, and resets the raster mode to one-bit direct (monochrome).

The data field must contain byte-aligned binary data, not ASCII. Unsupported or out-of-range values or other invalid configurations cause the entire command to be ignored and the data bytes to be discarded. Extra bytes are discarded. Value-field signs are ignored.

CRD provides raster configurations having one or more of the following characteristics:

- Horizontal resolution differs from vertical resolution.
- Color components may be specified at different resolutions.
- Each component may be specified at a multiple intensity level.
- Indexed raster data is separate from palette configuration.
- Direct raster data does not affect the palette configuration.
- Mixed indexed raster data and monochrome data for sleek top to bottom printing.

Reset (*EscE*), Simple Color (*Esc*r#U*), and Configure Image Data (*Esc*v#W*) override a CRD configuration.

Raster resolution is determined by the most recent CRD even if it is of another format, or by the Raster Resolution (*Esc*t#R*) command. A Raster Resolution command issued after CRD:

- Sets horizontal and vertical resolutions for each component to the specified resolution.
- Matches the strip height for each component to the new resolution.
- Establishes a pixel unit size for source raster height and width.
- Zeros the seed rows.
- Sets the number of levels for each component to 2.
- Has no effect on the number of expected components (raster indexing mode is unchanged).

The lowest CRD horizontal resolution determines the horizontal pixel size for the Source Raster Width (*Esc*r#S*), to which all the planes of all the rows for all components are zero-filled or clipped. Similarly, the lowest CRD vertical resolution determines the vertical pixel size for Source Raster Height (*Esc*r#T*), which defines raster area vertical height.

Missing data is zero-filled if Raster Resolution is changed or a Y Offset (*Esc*b#Y*) command is received after only part of the data defining a strip is sent.

NOTE: The Seed Row Source (*Esc*b#S*) command should not be used for CRD-configured raster data because the result is undefined.

As usual, the Transfer Raster commands (*Esc*b#W*, *Esc*b#V*) download raster data, and the Compression Method (*Esc*b#M*) command defines the compression rules.

CRD has different forms based on the format byte. Each form of CRD shares only the first byte, the format number, with the other forms of CRD.

NOTE: Format 1 was used only on DeskJet 520 and DeskJet 560C and is being obsoleted in favor of Format 2, which should be used on all future implementations.

Format 1   Complex Direct Planar Alternative

Format 1 provides configuration for raster transfers where:

- Color components may have different resolutions.
- Horizontal and vertical resolutions may be different.
- Color components may have different intensity levels.

NOTE: Format 1 was used only on DeskJet 520 and DeskJet 560C and is being obsoleted in favor of Format 2, which should be used on all future implementations.

The data for each color component is organized into strips containing one or more rows. The lowest vertical-resolution determines the vertical displacement of all the components in a strip; higher vertical-resolution components require more rows than lower vertical-resolution components to cover the same vertical displacement. The horizontal resolution of a component determines the number of bytes needed to define a row (e.g., 600 dpi rows contain 8 times as many bytes as 75 dpi rows). Higher horizontal and vertical resolutions must be multiples of lower horizontal and vertical resolutions, respectively. The CRD command is ignored for unsupported resolutions.

As shown below, the information specified for each color component consists of:

- horizontal resolution.
- vertical resolution.
- the number of levels of grayscale (continous tone) data.

Byte	15 (MSB) 8	7 (LSB) 0	Byte
0	Format=1 (UBYTE)	Number of components (UBYTE)	1
2	Horizontal Resolution for Component 1 (UINT16)		3
4	Vertical Resolution for Component 1 (UINT16)		5
6	Number of Levels for Component 1 (UINT16)		7
...	...		...
6(n-1)+2	Horizontal Resolution for Component n (UINT16)		6(n-1)+3
6(n-1)+4	Vertical Resolution for Component n (UINT16)		6(n-1)+5
6(n-1)+6	Number of Levels for Component n (UINT16)		6(n-1)+7

Byte 0:   Format

Value   =   1
Default   =   1
Range   =   1 to 5 (command is ignored for unsupported values)

Format 1 configures for direct monochrome, CMY, or KCMY data in planar format.

NOTE: Format 1, implemented only on DeskJet 520 and DeskJet 560C, should not be documented externally.

Byte 1   Number of Components

Value =   1   Monochrome (K)
       =   3   Three components (CMY)
       =   4   Four components (KCMY) (command is ignored for unsupported values)

Specifies the number of expected components. When the value is 3, the first component is cyan, the second magenta, and the third yellow. When the value is 4, the first component is cyan, the second magenta, the third yellow, and the fourth black. The raster transfer commands (*Esc*b#V*, *Esc*b#W*) must use the same component ordering, except in the four-components case, where the data is transferred black first, cyan, magenta, and finally yellow.

DEVICE NOTE: DeskJet 520 supports only 1 component.

Bytes 2&3, 8&9, etc   Horizontal Resolution

Value =   1 to 65535 (command is ignored for unsupported values)

Specifies the horizontal resolution of each component in pixels per inch (ppi) of the source pixel. The "horizontal" axis is determined by the current Raster Presentation (*Esc*r#/F*), Orientation (*Esc&l#O*), and Print Direction (*Esc&a#P*). Bytes 2 & 3 specify the horizontal resolution for component one, bytes 8 & 9 specify the horizontal resolution for component two, etc. The highest horizontal resolution must be a multiple of lower horizontal resolutions.

DEVICE NOTE: On DeskJet 520 and DeskJet 560C, a CRD command with a horizontal resolution of 600 and a vertical resolution of 300 will evoke Letter Quality and override Draft if in effect.

DEVICE NOTE: On DeskJet 520 and DeskJet 560C, all non-black CRD horizontal resolutions must match the resolution specified by the Raster Resolution command (*Esc*t#R*). Black CRD horizontal resolution must also match the Raster Resolution command (*Esc*t#R*) unless the black horizontal resolution is 600 and the resolution specified by the Raster Resolution command (*Esc*t#R*) is 300.

Bytes 4&5, 10&11, etc   Vertical Resolution

Value =   1 to 65535 (command is ignored for unsupported values)

Specifies the vertical resolution of each component of the source pixel in pixels per inch. The "vertical" axis is determined by the current Raster Presentation (*Esc*r#/F*), Orientation (*Esc&l#O*), and Print Direction (*Esc&a#P*). Bytes 4 & 5 specify the vertical resolution for component one, bytes 10 & 11 specify the vertical resolution for component two, etc. The highest vertical resolution must be a multiple of lower vertical resolutions.

DEVICE NOTE: On DeskJet 520 and DeskJet 560C, a CRD command with a horizontal resolution of 600 and a vertical resolution of 300 will evoke Letter Quality and override Draft if in effect.

DEVICE NOTE: On DeskJet 520 and DeskJet 560C, CRD vertical resolution must match the resolution specified by the Raster Resolution command (*Esc*t#R*).

Bytes 6&7, 12&13, etc   Number of Intensity Levels

Value =   2 to 255 (command is ignored for unsupported values)

Specifies the number of intensity or grayscale levels for each component. A level of 2 allows only two intensities (a pixel is either on or off). A level of 4 allows four intensities, etc. The lowest intensity is 0; the highest intensity is Number of Levels - 1.

Bytes 6 and 7 specify the number of levels for component one, bytes 12 and 13 specify the number of levels for component two, etc.

The number of planes expected for each row is the ceiling function of log base 2 of the number of levels. For example, three levels require two planes. The lowest order plane is transmitted first, and the highest order plane last for a given row.

DEVICE NOTE: DeskJet 520 and DeskJet 560C support only a level of 2.

Acceptable Combinations of Parameters for Format 1

A = Number of Components
B = Horizontal Resolution of Component 1
C = Vertical Resolution of Component 1
D = Number of Intensity Levels of Component 1
E = Horizontal Resolution of Component 2
F = Vertical Resolution of Component 2
G = Number of Intensity Levels of Component 2
H = Horizontal Resolution of Component 3
I = Vertical Resolution of Component 3
J = Number of Intensity Levels of Component 3
K = Horizontal Resolution of Component 4
L = Vertical Resolution of Component 4
M = Number of Intensity Levels of Component 4

DEVICE NOTE: Acceptable combinations of parameters for DeskJet 520

A	B	C	D
1	600	300	2
1	300	300	2
1	150	150	2
1	100	100	2
1	75	75	2

DEVICE NOTE: Acceptable combinations of parameters for DeskJet 560C

A	B	C	D	E	F	G	H	I	J	K	L	M
1	600	300	2									
1	300	300	2									
1	150	150	2									
1	100	100	2									
1	75	75	2									
3	300	300	2	300	300	2	300	300	2			
3	150	150	2	150	150	2	150	150	2			
3	100	100	2	100	100	2	100	100	2			
3	75	75	2	75	75	2	75	75	2			
4	600	300	2	300	300	2	300	300	2	300	300	2
4	300	300	2	300	300	2	300	300	2	300	300	2
4	150	150	2	150	150	2	150	150	2	150	150	2
4	100	100	2	100	100	2	100	100	2	100	100	2
4	75	75	2	75	75	2	75	75	2	75	75	2

Example 1 of Format 1

Assume the following CRD command is sent:

Format	=	1
Number of components	=	3
Horizontal resolution component 1	=	150
Vertical resolution component 1	=	150
Levels for component 1	=	2
Horizontal resolution component 2	=	150
Vertical resolution component 2	=	150
Levels for component 2	=	2
Horizontal resolution component 3	=	150
Vertical resolution component 3	=	150
Levels for component 3	=	2

In this example, the vertical displacement of all four strips is 1/150", since that is the displacement of the component with the lowest vertical resolution (150 ppi).

Component 1 consists of one row containing one plane of cyan. Only one row is needed at 150 ppi vertical resolution to cover 1/150" of vertical displacement. And only one plane is needed to encode two intensity levels ($log_2 2=1$).

Components 2 and 3 are encoded similarly. Component 2 consists of one row containing one plane of magenta; component 3 consists of one row containing one plane of yellow.

As shown below, components are sent in order from 1 to 3.

Component 1	row 1	*Esc*b#V*	150 ppi Cyan data
Component 2	row 1	*Esc*b#V*	150 ppi Magenta data
Component 3	row 1	*Esc*b#W*	150 ppi Yellow data

Example 2 of Format 1

Assume the following CRD command is sent:

Format	=	1
Number of components	=	4
Horizontal resolution component 1	=	300
Vertical resolution component 1	=	300
Levels for component 1	=	2
Horizontal resolution component 2	=	300
Vertical resolution component 2	=	300
Levels for component 2	=	2
Horizontal resolution component 3	=	300
Vertical resolution component 3	=	300
Levels for component 3	=	2
Horizontal resolution component 4	=	600
Vertical resolution component 4	=	300
Levels for component 4	=	2

In this example, the vertical displacement of all four strips is 1/300", since that is the displacement of the component with the lowest vertical resolution (300 ppi).

Component 1 consists of one row containing one plane of cyan. Only one row is needed at 300 ppi vertical resolution to cover 1/300" of vertical displacement. And only one plane is needed to encode two intensity levels ($\log_2 2 = 1$).

Components 2 and 3 are encoded similarly. Component 2 consists of one row containing one plane of magenta; component 3 consists of one row containing one plane of yellow.

Component 4 consists of one row containing one plane of 600 ppi horizontal by 300 ppi vertical black. As shown below, a 600 x 300 ppi plane contains twice the data of a 300 x 300 ppi plane.

Transfer Raster by Plane (*Esc*b#V*) is used to send all the planes of all the rows except the last plane of the last row within a strip, which uses Transfer Raster by Row/Block (*Esc*b#W*). The least significant plane of a row is sent first. Components are sent in the order: 4, 1, 2, 3.

Component 4	row 1	*Esc*b#V*	600 x 300 ppi Black data
Component 1	row 1	*Esc*b#V*	300 ppi Cyan data
Component 2	row 1	*Esc*b#V*	300 ppi Magenta data
Component 3	row 1	*Esc*b#W*	300 ppi Yellow data

20.4 Font Segment

Large Character Placement   *Esc * c # c/C*

Moves CAP to the specified column position on the current line

Value(#)   =   Absolute column position (+ or - is ignored)
Default   =   NA
Range   =   NA (values less than CAP move CAP one column to the right)

The value is based on the pitch of the current character set, or HMI if implemented.

The command provides device-independent large character positioning. Like the Horizontal Positioning command (*Esc&a#C*), starting location is specified by a column number. Horizontal spacing is based on the current font pitch, or on HMI if implemented.

Large Character Size   *Esc * c # m/M*

Defines the horizontal and vertical magnification of the standard character size.

Value(#)   =   Character size in # times the standard cell size)
Default   =   1
Range   =   1 to 28 (values less 1 are set to 1; values greater than 28 are set to 28)

Large Character Horizontal Size   *Esc * c # s/S*

> **NOTE:** This sequence has been reassigned to Symbol Set Code (Chapter 12).

Defines the horizontal magnification of the standard character size.

Value(#)   =   Horizontal character size in # times the standard cell size
Default   =   1
Range   =   1 to 28 (values less 1 are set to 1; values greater than 28 are set to 28)

The sequence provides for independent control of the horizontal and vertical magnification when used in conjunction with the Large Character Size command (*Esc*c#M*).

Large Character Tab   *Esc * c # n/N*

Moves CAP to the next horizontal tab stop on the current line. The location of the tab stop is determined by the current value of HMI, not by the horizontal magnification of the large characters. A value parameter is ignored.

Large Character X Offset   *Esc * c # x/X*

NOTE: This sequence has been reassigned to Picture Frame Horizontal Size (Decipoints) (Chapter 16).

Defines the relative horizontal offset in dots for the next large character.

Value(#) = Relative horizontal offset in dots (+ or - is ignored)
Default   = NA
Range   = 0 to 32767

The value field specifies the number of dots to the right to place the next large character symbol.

DEVICE NOTE: The dot spacing is 1/90 inch for the HP 2934A.

Large Character Print Data   *Esc * c <Text> z/Z*

Sends the large characters to the printer. Large Characters are sent to the device as an alphanumeric string enclosed in angled brackets. To specify an angled bracket in the data, a left angled bracket must precede it.

Upon termination of this escape sequence, the device prints all the buffered large character data at the selected magnification and position.

Control codes are treated as invalid characters.

DEVICE NOTE: In the HP 2934A, invalid characters in the alphanumeric string are ignored and replaced with a space.

Character Set Selection Control   *Esc & k # i/I*

Controls whether the current alternate character set is to be accessed via SI/SO control or via the eighth bit. If a 0 is received in the value field, SI/SO should be used. If a 1 is received, the eighth bit should be used. The requirement for this command does not exist, since the method the user desires is implied by the data he sends. If the eighth bit is a 1 or if a SO is received, the secondary character set should be used. If the eighth bit is a 0 and a SO has not been received (or a SI has), the primary character set should be used. The default access to the secondary character set should be via SI/SO. Note: This escape sequence is in conflict with the 8-bit character set standard.

Primary and Secondary Pitch Mode   *Esc & k # s/S*

Defines the pitch mode for both primary and secondary fonts.

Value	Pitch (Recommended Nominal)
0	Normal (10.0)
1	Double Wide (5.0)
2	Compressed (16.5 - 16.7)
3	Double Wide Compressed (8.25 - 8.35)
4	Elite (12.0)
8	Double Size (Double wide/high)

Default =   Defined by printer default set (generally 10 cpi)
Range =   0 to 8 (unavailable values are ignored)

This command defaults the HMI value to the width of the newly invoked font's SPACE.

Auto Centering and Justification   *Esc & l # j/J*

Centers or justifies print.

Value(#)	=	0	No centering or justification
	=	1	Right flush (optional)
	=	2	Centering
	=	3	Justification
Default	=	0	
Range	=	0 to 3	

Right flush moves the entire line to the right margin. If the line is wider than the margins, no change is made to the line. The right flush parameter is optional; centering and justification may be implemented without implementing right flush.

Centering positions the line between the left and right margins. If the line is wider than the margins, no centering is performed.

Justification takes a line of data and expands it from the position of the first printable character to the right margin. Leading spaces, tabs, or pen moves are not used in justification. The printer may stretch only the spaces between the words, or it may stretch the spacing between individual characters.

Character Set Default Control   *Esc & s # i/I*

Controls the action that should be taken when a request for an unavailable character set is received. If a 0 is received in the value field, a "blank" character set should be the default for an unavailable character set. If a 1 is received, the default for an unavailable character set should be to ignore the escape sequence. This command exists to provide compatibility with early terminal devices. The default should be to ignore a request for a character set which is not available.

Primary Font Placement (superior/inferior)   *Esc (s # u/U*
Secondary Font Placement (superior/inferior)   *Esc) s # u/U*

Specifies the font superior/inferior attribute.

Value(#)   =   0   Normal placement
           =   1   Inferior
           = -1   Superior
Default   =   0
Range   = -1 to 1 (unspecified values are ignored)

This command does not define character placement implementation: one printer may raise or lower a normal-height character by half a line; another may reduce the character size.

Print Enhancement (underline)   *Esc & k # e/E*

Determines whether underline works on a mode basis or a line-by-line basis

Value(#)   =   0   Line-by-line
           =   1   Mode
Default   =   1
Range   =   0,1 (command is ignored for other values)

Mode basis means that a state is active until turned off; line-by-line means the state turns off at the end of the print line.

Shift-In/Shift-Out Control   *Esc & k # f/F*

Determines whether the shift-in and shift-out control codes work on a mode basis or a line-by-line basis

Value(#)   =   0   Line-by-line (the state turns off at the end of the print line)
           =   1   Mode (the state is active until turned off)
Default   =   1
Range   =   0,1 (command is ignored for other values)

Font Quality (Primary)   *Esc (s # q/Q*
Font Quality (Secondary)   *Esc) s # q/Q*

Specifies font quality.

Value(#)   =   0   Data processing
           =   1   Near letter quality
           =   2   Letter quality
Default   =   Device dependent
Range   =   0 to 2.

This attribute determines printing quality, not the actual font representation. If the requested quality is unavailable, the closest fit is chosen.

DEVICE NOTE: The 256X series implements -1 as less than data processing quality.

20.5 Page Presentation Segment

Set Left Margin at Current Horizontal Position   *Esc 4*

Sets the left margin to the current active horizontal position.

Set Right Margin to Current Horizontal Position   *Esc 5*

Sets the right margin to the current horizontal position.

20.6 Device Control and Diagnostics Segment

I/O Status Request   *Esc ?*

This request is executed immediately when received (i.e., when the I/O checks for *Esc?* in the input data stream). *Esc?* is always acted upon, even in binary data; it must stripped from the input data stream if it appears in non-binary data. With an RS-232-C interface, *Esc?* must be followed immediately by a *DC1* to cause the printer to transmit the one-byte response, whose bits are defined as follows:

Bit		Definition
0 (LSB)	1	Paper out, manual feed, print error, operator and service calls.
1	1	Temporary printer busy (buffer full, warming up).
2	1	Printer off-line.
3	1	I/0 data error received since last status check.
		1) Parity error
		2) Framing error
		3) Overrun (input buffer or UART RAM)
4		Unused. Set to one.
5		Unused. Set to one.
6		Unused. Set to one.
7 (MSB)		Parity if seven-bit mode. Zero if eight-bit mode.

Primary Status Request   *Esc ^*

Requests the printer to return information on symbol set status, print modes, line density, power-on configuration modes, and hardware status. The information is in 4-bit groups encoded as characters from column 3 of the ASCII table. The number of characters varies, but the information is preceded by *Esc * and is terminated by CR-LF.

Primary Status: Request   *Esc^* (RS-232-C and RS-422 require a DC1 trigger)
                     Response   *Esc\\<six status bytes>CR-LF*

Return Model Number   *Esc * r # K*

Requests that the printer return its Hewlett-Packard model number followed by CR-LF. Using serial interface the printer must receive a DC1 trigger immediately after the terminator before sending the model number. The return model number escape sequence does not meet the general theory of zi (parameter character) and Zn (terminator character); only Zn is effective.

Return Model Number   *Esc * s # ^*

Requests that the printer return its Hewlett-Packard model number followed by CR-LF. Using serial interface the printer must receive a DC1 trigger immediately after the terminator before sending the model number. The return model number escape sequence does not meet the general theory of zi (parameter character) and Zn (terminator character); only Zn is effective.

QMS Magnum-V Enable/Disable   *Esc * t # F*

Enables or disables the QMS Magnum-V language interpreter.

Value(#)   =   0   Disable QMS Magnum-V
           =   1   Enable QMS Magnum-V
Default   =   0
Range   =   0,1

Transfer to the On-Line State   *Esc n*

Causes no action in a printer. This sequence should not be used. A more workable solution is for the printer to power up in the same off/on-line state it was in when it powered down if the machine has nonvolatile RAM, or to power up in the on-line state for recoverability.

Transfer to the Off-Line State   *Esc o*

Transfers the printer to the off-line state if presently on-line. This escape sequence can be used by devices which require operator intervention. Upon receipt of this escape sequence, all buffered data should be processed.

Head View Enable/Disable   *Esc & k # v/V*

Controls head view mode. When head view is enabled, the print head moves away from the current active position so the most recent print can be viewed when not actively printing. Printers with a keyboard interface require this feature.

```
Value(#) = 0   Enables head view mode
         = 1   Disables head view mode
Default  = 1
Range    = 0,1
```