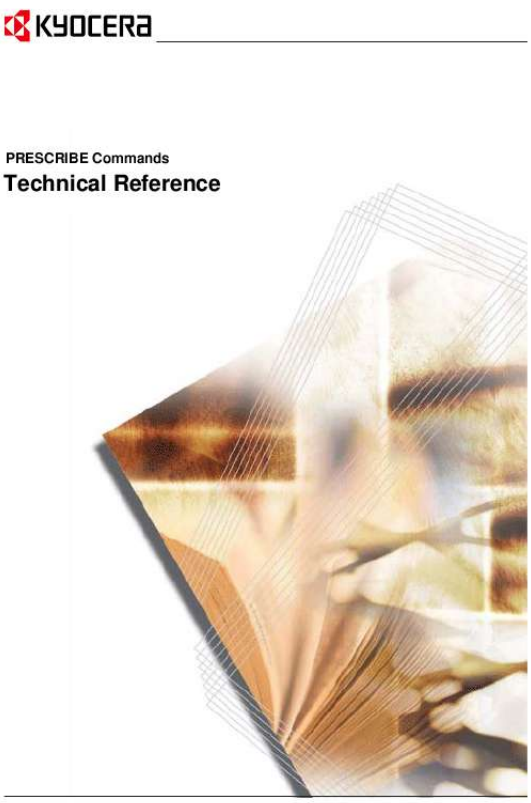

Manual abstract:

Copying or other reproduction of all or part of this manual, the computer programs, or any other copyrightable subject matter without the prior written consent of Kyocera Mita Corporation is prohibited. Any copies made of all or part of this manual, the computer programs, or any other copyrightable subject must contain the same copyright notice as the material from which the copying is done. The information in this manual is subject to change without notification. Additional pages may be inserted in future editions. The user is asked to excuse any omissions or errors in the present edition. No responsibility is assumed if accidents occur while the user is following the instructions in this manual. No responsibility is assumed for defects in the printing system's firmware. Regarding Tradenames PRESCRIBE is a registered trademark of Kyocera Corporation. KPDL is a trademark of Kyocera Corporation. Diablo 630 is a product of Xerox Corporation.

IBM Proprinter X-24E is a product of International Business Machines Corporation. Epson LQ-850 is a product of Seiko Epson Corporation. HP LaserJet, Hewlett-Packard, PCL, and HP-GL/2 are registered trademarks of Hewlett-Packard Company. Other product names and company names that appear in this manual are trademarks or registered trademarks of their respective owners. The Kyocera printing systems use PeerlessPrint5 to provide the HP LaserJet compatible PCL5 language emulation.

PeerlessPrint5 is a trademark of The Peerless Group, Redondo Beach, CA 90278, U.S.A. © Copyright 2005 - 2008 KYOCERA MITA Corporation All rights reserved Revision 4.51 2008.

6 Typeface Trademark Acknowledgement All resident fonts in the printing system are licensed from Bitstream Inc. and Agfa corporation. For font license information for each model, refer to the User's Manual. Helvetica, Palatino and Times are registered trademarks of Linotype-Hell AG. Century Schoolbook, Stymie, and CooperBlack are trademarks of Kingsley-ATF Type Corporation. ITC Avant Garde Gothic, ITC ZapfChancery, ITC ZapfDingbats, ITC Souvenir, ITC Benguiat, and ITC Bookman are registered trademarks of International Typeface Corporation. Revue is a trademark of Esselte Pendaflex Corporation in the U.S., Letraset Canada Ltd. in Canada, and Esselte Letraset Ltd.

elsewhere. Agfa Japan License Agreement Guidelines for the Resident Fonts 1. 2. Software shall mean the digitally encoded, machine readable, scalable outline data as encoded in a special format as well as the UFST Software. You agree to accept a non-exclusive license to use the Software to reproduce and display weights, styles and versions of letters, numerals, characters and symbols ("Typefaces") solely for your own customary business or personal purposes at the address stated on the registration card you return to Agfa Japan. Under the terms of this License Agreement, you have the right to use the Fonts on up to three printing systems. If you need to have access to the fonts on i more than three printing systems, you need to acquire a multi-user license agreement which can be obtained from Agfa Japan. Agfa Japan retains all rights, title and interest to the Software and Typefaces and no rights are granted to you other than a License to use the Software on the terms expressly set forth in this Agreement. 3. To protect proprietary rights of Agfa Japan, you agree to maintain the Software and other proprietary information concerning the Typefaces in strict confidence and to establish reasonable procedures regulating access to and use of the Software and Typefaces.

You agree not to duplicate or copy the Software or Typefaces, except that you may make one backup copy. You agree that any such copy shall contain the same proprietary notices as those appearing on the original. This License shall continue until the last use of the Software and Typefaces, unless sooner terminated. This License may be terminated by Agfa Japan if you fail to comply with the terms of this License and such failure is not remedied within thirty (30) days after notice from Agfa Japan. When this License expires or is terminated, you shall either return to Agfa Japan or destroy all copies of the Software and Typefaces and documentation as requested.

You agree that you will not modify, alter, disassemble, decrypt, reverse engineer or decompile the Software. Agfa Japan warrants that for ninety (90) days after delivery, the Software will perform in accordance with Agfa Japanpublished specifications, and the diskette will be free from defects in material and workmanship. Agfa Japan does not warrant that the Software is free from all bugs, errors and omissions. THE PARTIES AGREE THAT ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE AND MERCHANTABILITY, ARE EXCLUDED. 8.

Your exclusive remedy and the sole liability of Agfa Japan in connection with the Software and Typefaces is repair or replacement of defective parts, upon their return to Agfa Japan. IN NO EVENT WILL AGFA JAPAN BE LIABLE FOR LOST PROFITS, LOST DATA, OR ANY OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES CAUSED BY ABUSE OR MISAPPLICATION OF THE SOFTWARE AND TYPEFACES. 9. New York, U.S.A. law governs this Agreement. 10. You shall not sublicense, sell, lease, or otherwise transfer the Software and/or Typefaces without the prior written consent of Agfa Japan. 11.

Use, duplication or disclosure by the Government is subject to restrictions as set forth in the Rights in Technical Data and Computer Software clause at FAR 252-227-7013, subdivision (b)(3)(ii) or subparagraph (c)(1)(ii), as appropriate. Further use, duplication or disclosure is subject to restrictions applicable to restricted rights software as set forth in FAR 52.227-19 (c)(2). 12. YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, UNDERSTAND IT, AND AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS. NEITHER PARTY SHALL BE BOUND BY ANY STATEMENT OR REPRESENTATION NOT CONTAINED IN THIS AGREEMENT. NO CHANGE IN THIS AGREEMENT IS EFFECTIVE UNLESS WRITTEN AND SIGNED BY PROPERLY AUTHORIZED REPRESENTATIVES OF EACH PARTY. BY OPENING THIS DISKETTE PACKAGE, YOU AGREE TO ACCEPT THE TERMS AND CONDITIONS OF THIS AGREEMENT. 4. 5.

6. 7. ii Introduction This manual contains information needed to use the firmware features provided by the Kyocera printing system. Among these features is PRESCRIBE, a highly accessible, human-readable command language that makes it easy for programmers to take full advantage of the printing system's capability. The PRESCRIBE command language allows to: · · · · extensive manipulation of fonts and character code tables use the ability to draw objects by constructing and manipulating paths including ellipses and round boxes, etc.

execute macros including carbon-copy macro control external optional units (feeders, etc.) You can access the features of PRESCRIBE from any of the seven emulation modes. These modes include: · · · · · · · Hewlett-Packard LaserJet emulation Hewlett-Packard HP 7550A (plotter) emulation IBM Proprinter X24E (24-pin dot matrix printer) emulation Epson LQ-850 (24-pin dot matrix printer) emulation Diablo 630 emulation generic line printer emulation KPDL (Apple LaserWriter II NTX (NT) emulation) [an option on some models] iii About the Technical Reference manual The Technical Reference manual is organized into eight chapters.

The first four chapters of this manual constitute an tutorial introduction to PRESCRIBE. The rests mainly concern advanced utilities and setups: Chapter 1 Introduction to PRESCRIBE introduces some basic concepts of PRESCRIBE.

Chapter 2 Graphics Tutorial outlines the graphic handling features of PRESCRIBE. Chapter 3 Macros introduces program macros, a concept that makes it easy to define sequences of PRESCRIBE commands, then call them repeatedly whenever they are needed. Chapter 4 Fonts provides how to manage font selection and font samples. Later, Chapter 5 Barcodes explains the barcode printing capabilities of the command language. Chapter 6 Permanent Parameters explains how to reprogram the printing system's firmware for customization. Chapter 7 Emulation gives notes on the printing system's various emulation modes. An Index is also provided at the end of this manual. Notice Most PRESCRIBE commands operate in the same way on all of these models. However, on particular models, some commands are irrelevant. Model-dependent differences are noted at the pertinent locations in this manual.

Conventions · · italic is used for emphasis and also refers to a related chapter or section in this manual or another related document. fixed-pitch means text or commands that you must type exactly as it appears. iv Table of Contents General Infomation .....

.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
... i Introduction..
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

.....
.....
.....
.....

....iii Introduction to PRESCRIBE Format of PRESCRIBE Commands .....
.....
.....
.....

.....
.....
.....
.....

.....
.....

user. In contrast, PRESCRIBE commands are made of ordinary characters that you can type in yourself and see on the computer screen. This makes it easy for you to customize printing and add features that may not be supported by your application. This chapter presents an introduction to PRESCRIBE starting with an explanation of the commands by which you start and exit PRESCRIBE. It is followed by an introduction to some basic concepts of PRESCRIBE, then a discussion of the command format and command parameters. Chapter 1 Introduction to PRESCRIBE Entry and Exit The printing system can be thought of as having a multiple personality. When its power is switched on, it performs the normal printing system functions of printing out files and other data. Application software can control the printing system using one of the seven emulations. When the printing system uses an emulation, it is said to be printing in emulation mode. PRESCRIBE is an additional mode of operation in which the printing system understands data it receives not as text to be printed, but as commands to be executed.

The PRESCRIBE mode is available at any time during operation from any emulation mode. The initializing string that takes the printing system from the usual text-printing mode into the PRESCRIBE mode is !R!. The command that returns it from the PRESCRIBE mode to the emulation mode is EXIT;. These transitions are diagramed in the figure below. Figure 1.

1. Mode Transitions The printing system's emulation mode can be permanently set by the FRPO (Firmware RePrOgram) P1 command. See Chapter 7 for details. The printing system is factory-set to emulate the Hewlett-Packard LaserJet. The example below shows how these transitions can be used in a file. The lines beginning with !R! are PRESCRIBE commands. Note how each block of commands begins with !R! and ends with EXIT;. These sections are not printed; instead, they set margins, select three different fonts, and draw a box around one line of text. The remainder of the file consists of ordinary text, and is printed out as shown in the figure on the next page. 1-2 Figure 1. 2. Text Including PRESCRIBE Commands !R! RES; SLM 1; STM 1; SPD 0.03; FTMD 13; SFNT "Helvetica-Bd"; EXIT; WELCOME TO WINDFALL NATIONAL PARK !R! SFNT "Times-Rom"; EXIT; The park entrance is located in the rolling hills of the Northern Woods, a forested area abundant in deer, elk, squirrel, rabbit, opossum, lynx, wolf, and other wildlife. It is the park's most popular area, featuring excellent trails and campsites for hiking and backpacking. !R! BOX 3.

4, 0.55; FSET 1s5B; EXIT; Hikers should avoid this area during hunting season. !R! SFNT "Times-Rom"; EXIT; Hikers in search of high-altitude adventure will find Mt.Baker a rewarding... Figure 1. 3. PRESCRIBE Example The previous example contains five sequences of PRESCRIBE commands. The basic configuration of a PRESCRIBE command sequence is: !R! command; command; .

..; command; EXIT; There is a limit to the number of commands you can include between the initial !R! and the final EXIT;. The initial !R! must be followed by a space, and each command must end with a semicolon. The use of PRESCRIBE commands in document files is conditional on the behavior of your word processing software.

Some word processing programs add control codes that interfere with PRESCRIBE. If you cannot control software in this way, try using a non-word processing mode (ASCII text function, for example) of the software. 1-3 Note Chapter 1 Introduction to PRESCRIBE Format of PRESCRIBE Commands The basic format of a PRESCRIBE command is: · · · or · · · · (command name) parameter, ...

, parameter; The command names generally consist of three or four letters. In most commands, the parameters must be followed by commas. The last parameter is always followed by a semicolon. Some commands (RES, for example) have no parameters. In this case, the command should be followed immediately by a semicolon (RES;). The length of a single PRESCRIBE command is limited to 255 characters, from the first letter of the command name through the final semicolon. Commands longer than 255 characters are not executed. Spaces, carriage return codes, and line feed codes are generally ignored in PRESCRIBE command sequences. These characters are not generally counted in the command length. (Exception: Spaces are not ignored in quoted character strings.

) To improve readability, place at least one space before each command or place each command on a separate line. Basic Concepts This section discusses a few basic concepts concerning how the printing system prints on the page. These concepts are: · · · · · · · Edge limits Margins Coordinate systems Text positioning/Character spacing Paths Logical page and physical page Page orientation and direction Edge Limits The printing system cannot place print on the outside edges of the paper. The edge limits to which printing is possible are located 5 mm inside the edges of the paper; or 6 mm (5mm in landscape orientation) from the left edge and 4 mm from the top of the paper in HP LaserJet emulation. Refer to the figure Edge Limits and Margins on page 5. The edge limits adjust automatically to the size of the paper cassette (although not to the size of manually fed paper). The edge limits can also be set to various standard sizes by the SPSZ (set paper size) command, or by the equivalent HP LaserJet commands. In HP LaserJet emulation, the edge limits slightly vary according to the page orientation (as also shown in Edge Limits and Margins on page 5). Margins The top and left margins are set in centimeters or inches in relation to the top and left edge limits of the page. The bottom and right margins can also be set as a distance from the top and left edge limits, or they can be set in terms of page width, page length, or lines per page.

1-4 Basic Concepts Figure 1. 4. Edge Limits and Margins Edge limits in HP emulation Portrait Orientation Edge limits in HP emulation Landscape orientation When the printing system passes the bottom margin while printing text, it prints the page and feeds to the next page. Spacing is carried over, so if the bottom margin does not occur at an exact number of lines, excess space is printed at the top of the next page. If you are using word-processing software that sets the margins automatically, you should not set them with PRESCRIBE commands.

Page Orientation and Direction The term page orientation refers to the direction in which text is placed on the page. In a vertical direction, it would be called portrait and a horizontal direction would be called landscape. The term print direction, which follows this section, refers to the orientation of the logical page's coordinate system with respect to the current page orientation. 1-5 Chapter 1 Introduction to PRESCRIBE Page Orientation Changing the page orientation automatically adjusts the margins so that they remain the same distance from the four edges of the paper. If the printing system cannot make these margin settings (for example, if the left margin would be to the right of the right margin), it sets the margins to the edge limits.

Fonts are automatically rotated to match the current orientation. Figure 1. 5. Page Orientations Portrait Orientation Top margin Right margin Left margin Left margin Right margin Bottom margin Bottom margin Print Direction The print direction can be modified in 90° increments. These page orientations are referred to as portrait, landscape, reverse portrait, and reverse landscape. Changing the print direction rotates the page coordinate system in the same manner as changing the page orientation. However, in this case, portrait refers to the print direction in which the axes of the coordinate system are oriented in the same direction as for the currently selected page orientation. Changing the print direction also changes the margins to maintain the same printable area as prior to the change. The current position (the physical location in which the next character will be printed) and its coordinate values remain the same as in the previous print direction. Changing the print direction also changes the orientation of any subsequent raster graphics and PRESCRIBE vector graphics.

However, it does not affect the orientation of any subsequent HP-GL/2 graphics. (HP-GL/2 graphics can only be rotated with the HP-GL/2 RO command or the LaserJet orientation command.) 1-6 Basic Concepts Figure 1. 6. Print Direction Portrait print direction Landscape print direction Top margin Current Point Bottom margin Current Point Left margin Top margin Right margin Left margin Right margin Reverse landscape print direction Reverse portrait print direction Current point Top margin Current point Left margin Top margin Bottom margin Right margin Right margin Coordinate Systems With PRESCRIBE, positions on a page are described in terms of X and Y coordinates. The origin of the coordinate system (the position at which X and Y both equal 0) is located at the intersection of the top margin and the left margin. Values of X greater than 0 indicate positions to the right of the origin, and values of Y indicate positions below the origin. See the figure on page 10. When the top and left margins are changed, the physical position of the origin changes accordingly. Text Positioning The printing system always keeps track of its current position on the page.

The current position can be thought of as a cursor that moves as data is printed. At any instant, the 1-7 Left margin Bottom margin Bottom margin Chapter 1 Introduction to PRESCRIBE cursor indicates where the next character will be printed or the next graphics will be drawn. (The printing system does not have separate cursors for text and graphics.) Text and graphics can be positioned at arbitrary locations on the page by moving the cursor with positioning commands (MAP, for example). Figure 1.

7. Text Positioning TITLE !R! BOX 1, 1; MRP 2, 1; EXIT; LABEL Landscape Orientation Carriage Return MRP 2, 1; BOX 1, 1; Space after EXIT; Cursor ends here Character Spacing Each character is printed within an individual cell as shown below.

The character sits on a line called the baseline. Characters such as y descend below the base line. In some character fonts, all the character cells are the same size, so the number of character positions per inch is fixed.

In other fonts, the size of character cells is proportional to the size of characters. These proportional fonts produce text that is easier to read. However, in order to align the right margin, you must use software that supports the printing system's proportional spacing. 1-8 Basic Concepts Figure 1. 8. Character Spacing Font height Baseline Paths A path is a set of straight and curved line segments. Paths can be open, as in the case of lines, or closed, as in the case of rectangles, circles, or any fully enclosed area of any shape. The segments may be connected with one another, or they may be disconnected. Further, a path may contain multiple closed subpaths, representing several areas, and they may intersect themselves in arbitrary ways. Paths can be used to draw lines and curves or specify boundaries of filled areas, including the outline of a character.

Paths are explained more fully in section Path Mode Graphics on page 16. Logical Page and Physical Page The logical page defines the limits of the coordinates within which text and graphics can be located. There are two types of logical page, as shown in the following figure. The standard mode logical page imposes limits on specifiable coordinates. The coordinates have no limitations for the path mode logical page. 1-9 Chapter 1 Introduction to PRESCRIBE Figure 1. 9. Logical Page and Physical Page Standard Mode x (0,0) = Logical page y Path Mode -y -x x y With the standard mode logical page, any position specification that lies outside of the logical limits is automatically adjusted to bring it within the limits. For example, the page on the upper left in the preceding figure shows what happens if you attempt to draw a diagonal line from below the bottom edge limit to a point to the right of the right edge limit when the standard mode logical page is used. The fine line represents the line as specified by the user; the thick line shows what is actually drawn by the printing system.

With the path mode logical page, coordinates are not adjusted even if they fall outside of the edge limits. In this case, as shown in the lower right page in the preceding figure, the line is defined by the specified starting and ending points, but parts falling outside of the edge limits are clipped. Command Parameters Numeric Parameters Many of the PRESCRIBE commands use number values to specify parameters. For example, numbers are used to specify distances in inches, centimeters, points, or dots. Negative numbers are also allowed.

1-10 Command Parameters For computer code values beyond four decimal places, the fifth and subsequent decimal places are ignored. Examples: Number output by computer 1234.1234 -1234.1234 0.123456 Number used by printing system 1234.

1234 -1234.1234 0.1234 Some commands have angle parameters. Angles are specified in degrees. (The printing system does not recognize radians). The printing system rounds off all angles to the nearest integral degree. Only angles in the range from -360 degrees to 360 degrees are recognized. Angles less than -360 degrees are ignored, and angles greater than 360 degrees are treated as the remainder of the angle divided by 360. Examples: Angle output by computer 90 -90 90.4 90.

5 -400 Angle used by printing system (degrees) 90 -90 90 91 Ignored The printing system does not accept the exponential notation used in some computer languages. For example, do not specify 1E-3 instead of 0.001. Character Strings PRESCRIBE text-printing commands have parameters that consist of character strings. A character string is any string of characters enclosed by quotation marks or apostrophes, such as shown in the example below. TEXT 'You are about to enter PRESCRIBE.'; PRESCRIBE allows character strings to be enclosed in either single quotation marks (apostrophes) or double quotation marks. The following example has exactly the same meaning as the one above. TEXT "You are about to enter PRESCRIBE."; The beginning of a character string is recognized when the first single or double quotation mark appears.

If the beginning quotation mark is a single quotation mark, the string does not end until the next single quotation mark. If the beginning quotation mark is a double quotation mark, the string does not end until the next double quotation mark. Whatever comes in the middle of a character string, including commas, semicolons, and even PRESCRIBE command names, is recognized as part of the character string, and not as part of the PRESCRIBE command language. For example, the expression EXIT; in the following string is just text; it does not cause the printing system to exit from the PRESCRIBE mode. TEXT 'NO EXIT; NO RETURN.

'; When the string itself contains one type of quotation mark, the quotation mark must be enclosed in quotes of the other type. Here are two examples: 1-11 Chapter 1 Introduction to PRESCRIBE TEXT "You're about to enter PRESCRIBE."; TEXT ' " " " '; In the first command above, the character string starts with a double quotation mark. The printing system therefore expects the string to end with a double quotation mark, and regards the apostrophe in the word You're as an ordinary character, not as the string terminator. Similarly, the double quotation marks in the second command above are recognized as ordinary characters, not as string terminators.

Since an apostrophe or quotation mark can start a character string anywhere in a PRESCRIBE command sequence, it is important not to start character strings unintentionally. The following examples demonstrate incorrect use of apostrophes and double quotation marks. Incorrect: !R! CMNT Don't leave stray apostrophes; EXIT; Incorrect: !R! CMNT The symbol " means inches; EXIT; In both of the above cases, the printing system assumes that the expression EXIT; is part of a character string started by the preceding apostrophe or quotation mark, and fails to exit the PRESCRIBE mode. The correct way to write these comments is: Correct: !R! CMNT "Don't leave stray apostrophes"; EXIT; Correct: !R! CMNT 'The symbol " means inches'; EXIT; Character strings must not exceed the 255-character limit on total command length. If a character string exceeds this limit, the printing system terminates it forcibly and begins looking for the next PRESCRIBE command. Upper and Lowercase Letters Regarding upper and lowercase characters, PRESCRIBE follows the same rule as many computer programming languages: it discriminates case inside character strings and ignores it elsewhere. You can type command names in upper or lowercase. Correct: !R! TEXT 'A'; CIR 1; EXIT; Also correct: !R! text 'A'; cir 1; exit; Also correct: 1-12 Command Parameters !R! Text 'A'; Cir 1; Exit; Each of these commands prints the capital letter 'A' inside a circle.

In the printout shown above, the unit is centimeters. The reason that the letter 'A' is off center in the circle is that the cursor is not located at the center of the circle, but at left corner of the letter 'A'.

The command !R! TEXT 'a'; CIR 1; EXIT; prints a lowercase a because the letter occurs inside a character string. The sole exception to upper and lowercase usage in PRESCRIBE commands occurs with the initializing !R! command. This command must always use an uppercase R. The printing system will not enter the PRESCRIBE mode in response to !r!. In this manual, PRESCRIBE commands are printed in upper-case for readability. Outside of PRESCRIBE mode, the printing system always distinguishes between uppercase and lowercase letters and prints exactly what is sent. Special Parameters Some PRESCRIBE commands use unquoted strings of characters as parameters. Examples for these are the FSET (change current font set by characteristic) command and the CSET (change symbol set by symbol-set ID) command. (See Chapter 4 for a detailed explanation of how these commands are used to select fonts.) The FSET and CSET commands use parameters that closely resemble the command parameters used for font control in Hewlett-Packard's printer control language. For example, the PRESCRIBE command FSET 0p12h12v0s0b6T; selects the font whose characteristics most closely matches the following font parameters: · · · · · Monospaced font (0p) Character spacing of 12 characters/inch (12h) Character height of 12 points (12v) Upright style (0s) Medium weight (0b) LetterGothicBM12-Roman typeface (6T) In Hewlett-Packard's PCL, the corresponding command would be ESC(s0p12h12v0s0b6T 1-13 Chapter 1 Introduction to PRESCRIBE Similarly, the PRESCRIBE command CSET 11U; designates use of the PC-8 Danish/Norwegian symbol set. The corresponding Hewlett-Packard PCL command is ESC(11U. 1-14 Chapter 2 Graphics Tutorial PRESCRIBE provides a wide variety of graphics operators, allowing you to easily construct and print almost any imaginable shape or pattern. This chapter introduces the various graphics concepts of PRESCRIBE, and illustrates how to use many of its graphic functions. It defines standard graphics mode, path mode graphics, and raster graphics.

It explains how to use predefined fill patterns, how to define your own fill patterns, and introduces ways in which you can change the print model, the rules that determine the manner in which patterns and images are rendered on the paper. Chapter 2 Graphics Tutorial Standard Graphics The standard graphics mode provides a number of operators for constructing a variety of filled shapes and lines. Using standard mode graphics, you can: · · · · Draw lines of any desired width Draw circles and rectangles Draw a variety of filled shapes, including boxes and arcs Draw pie charts This is referred to as the standard graphics mode because it is a standard feature of all versions of PRESCRIBE. Drawing Lines PRESCRIBE provides a number of Draw to commands for drawing lines in both standard and path modes. These include: Draws a line to an absolute position in a Cartesian coordinate system whose origin (0,0) is at the intersection of the left and top margins.

DZP (draw to zero-relative Draws a line to an absolute position in a Cartesian coorposition) dinate system whose origin (0,0) is at the intersection of the left and top edge limits of the paper. DRP (draw to relative position) Draws a line to a position specified as a horizontal and vertical displacement from the current cursor position. DRPA (draw to relative Draws a line to a position that is specified as a distance and position specified by angle) angle from the current cursor position. Examples of these commands are given in the sections that follow. DAP (draw to absolute position) Lines to Absolute Position Begin with a simple task such as drawing a line between two arbitrary points on a page. Use the MAP and DAP commands to specify positions relative to the top and left margins. This task has several distinct steps: selecting a line width, determining the starting point of the line, and determining the end point of the line. The following command sequence demonstrates this process. !R! RES; STM 0.5; SLM 0.

5; SPD 0.01; MAP 0.5, 1; DAP 2, 0.5; PAGE; EXIT; 2-2 Figure 2. 1. Result of Draw Commands: Absolute Lines Edge limits Margins DAP 2, 0.5; MAP 0.5, 1; The initial !R! command switches the printing system to the PRESCRIBE mode. Remember that this command must always precede each sequence of PRESCRIBE commands. The RES (RESet) clears the current page from printing system memory and re-establishes the printing system's permanent defaults. Although you would not include this command in every sequence of PRESCRIBE commands, we include it in this example to ensure consistent results. As a standard practice, include the RES at the beginning and end of each job. The STM and SLM set both the top and left margins to 0.5 inches (1.27 centimeters). The SPD (Set Pen Diameter) command determines the thickness of lines. In the standard graphics mode, this setting determines the thickness of all lines drawn after the command is issued. In this example, the line width is set to 0.01 inches. The starting point of the line is established with the MAP (Move to Absolute Position) command.

This command moves the cursor to a point that is a specified distance from the top and left margins. In this example, the point specified is 0.5 inches from the left margin and 1 inch from the top margin. If the margins are changed, the position specified by MAP also changes correspondingly. On the next line of the program, the DAP (Draw to Absolute Position) command draws a line from the starting position to the point 2 inches from the left margin and 0.5 inches from the top margin. Finally, PAGE; prints out the page, allowing us to look at the result of our work. Zero-relative Lines The line draw example below uses some new commands to draw another line. !R! RES; SPD 0.01; MZP 0.

5, 1; DZP 2, 0.5; PAGE; EXIT; 2-3 Chapter 2 Graphics Tutorial Figure 2. 2. Result of Draw Commands: Zero-relative Lines Edge limits DZP 2, 0.5; MZP 0.5, 1; The first two lines switch the printing system to the PRESCRIBE mode, reset printing system parameters, and set the line width to 0.01 inch. On the third line, the MZP (Move to Zero-relative Position) differs from the MAP (Move to Absolute Position) command in one respect: the position specified is in relation to the top and left edge page limits of the page, rather than in relation to the top and left page margins. MZP moves the cursor to the point that is 0.5 inches from the left edge limit and 1 inch from the top edge limit.

Similarly, on the next line, DZP (Draw to Zero-relative Position) draws a line from the starting position to the point 2 inches from the left edge limit and 0.5 inches from the top edge limit.

Relative Lines Another way to specify positions is in relation to the current cursor position. The following command sequence provides an example. !R! RES; SPD 0.

01; MRP 2, 1; DRP -1.5, -1; MRP 2, 1; DRP -1.5, -1; MRP 2, 1; DRP -1.5, -1; PAGE; EXIT; 2-4 Figure 2. 3.

Result of Draw Commands: Relative Lines Edge limits In this command sequence, the PRESCRIBE mode begins with the !R!, resets the printing system defaults to permanent settings with RES;, and establishes a pen width of 0.01 inches with the SPD 0.01;. The MRP (Move to Relative Position) and DRP (Draw to Relative Position) specify positions in relation to the cursor's current position. When the command sequence starts, the cursor is located at the intersection of the left and top margins. The command MRP 2, 1; on line 3 moves the cursor 2 inches to the right of its current position, and down 1 inch from its current position. Then the command DRP -1.5, -1; draws a line from that point to a point 1.5 inches to the left of the cursor position and 1 inch above it. The cursor winds up 0.

5 inches to the right of the point where it started. Lines 5 to 8 repeat the move-and-draw sequence two more times. This produces three parallel lines, as shown in the figure on the previous page. Lines in Terms of Angles Until now, all of our examples have specified positions in terms of Cartesian (X,Y) coordinates. This example illustrates drawing lines of specified lengths and angles. !R! RES; SPD 0.01; MZP 5, 4; DRPA 2, 149; DRPA 2, 221; DRPA 2, 293; DRPA 2, 365; CMNT Equivalent to 5 degrees; DRPA 2, 437; CMNT Equivalent to 77 degrees; PAGE; EXIT; 2-5 Chapter 2 Graphics Tutorial Figure 2. 4. Result of Draw Commands: Lines in Angles The first two lines of this command sequence initiates the PRESCRIBE mode, resets printing system defaults to permanent settings, and sets the line width to 0.01 inches.

Then the MZP command on line 3 moves the cursor to a point 5 inches to the right of the left edge limit and 4 inches below the top edge limit. Next, DRPA 2, 149; on line 4 draws a line two inches long at an angle of 149 degrees. The angle is measured clockwise from the vertical axis. The subsequent DRPA commands draw additional 2-inch lines at angles that increase in increments of 72 degrees. As indicated by the CMNT (CoMmeNT) commands, angles that exceed 360 degrees are equivalent to the remainder of division of the angle by 360.

Drawing Boxes and Circles PRESCRIBE provides two commands especially for drawing boxes (BOX command) and circles (CIR command). Drawing Boxes The BOX (draw box) command draws a box of a specified width and height. As with the line drawing commands, the thickness of the line used to draw the box is determined by the SPD (set pen diameter) command. The following command sequence draws a box. !R! RES; UNIT C; SPD 0.

1; MZP 3, 3; BOX 3, 4; PAGE; EXIT; 2-6 Figure 2. 5. An Example of a Box Line 1 places the printing system in the PRESCRIBE mode and resets printing system parameters. The UNIT C; command on the second line sets the unit of measurement to centimeters, and the SPD (Set Pen Diameter) command on line 3 sets the line width to 0.1 centimeters. (If you omit these two commands, the printing system will print using the default unit, inches; and the default line width, 3 dots.) Next, the MZP command on line four moves the cursor to the point that is 3 centimeters to the right of the left edge limit and 3 centimeters below the top edge limit. This is the starting point from which the box is drawn. On line 5, BOX 3, 4; draws a box with a width of 3 centimeters and a height of 4 centimeters. The position of the box with respect to the cursor depends on the positive or negative value specified for width and height.

The box is drawn to the right of the cursor if width is positive, and to the left of the cursor if width is negative. Similarly, the box is drawn below the cursor if height is positive, and above the cursor if height is negative. This relation is illustrated in the figure that follows. By default, the position of the cursor is not affected by this command. However, you can also specify an option parameter to make the cursor move to an adjacent or diagonally opposite corner of the box, down by one text line, or to the left margin on the next text line. The following figure shows some examples. 2-7 Chapter 2 Graphics Tutorial Figure 2. 6. Cursor Positioning Options BOX 4, 2, H; Moves the cursor to the horizontally adjacent corner -Y -X Cursor position X Y BOX 4, 2, V; Moves the cursor to the vertically adjacent corner BOX 4, 2, E; Moves the cursor to the diagonally opposite corner BOX 4, 2, L; Moves the cursor down one line Left margin BOX 4, 2, N; Moves the cursor to the beginning of the next line (to the left margin) 2-8 Drawing Circles The CIR (draw circle) command draws a circle of a specified radius using the line thickness set by the SPD (set pen diameter) command. The circle drawn is centered on the current cursor position; the position of the cursor remains unaffected.

See the following example: !R! RES; UNIT C; SPD 0.1; MZP 8, 8; CIR 1; CIR 2; CIR 3; PAGE; EXIT; Figure 2. 7. Circles Lines 1, 2 and 3 start PRESCRIBE mode, reset the printing system to its default parameters, establish the unit of measurement as centimeters, and set the line width to 0.1 centimeters. Next, the MZP command moves the cursor to the point that is 8 centimeters to the right of the left edge limit and 8 centimeters below the top edge limit. Lines 5, 6, and 7 draw three circles with radii of 1, 2, and 3 centimeters. Drawing Filled Shapes The standard graphics mode provides two types of filled shapes: arcs and blocks. Such shapes are filled with one of the printing system's predefined patterns, or with a user defined pattern. Filled areas of other shapes can be printed using path mode graphics.

For details, see the explanation in section Path Mode Graphics on page 16. 2-9 Chapter 2 Graphics Tutorial A filled block consists simply of a rectangle of any desired dimensions. A filled arc is an area enclosed by an arc segment and the line segments extending from the ends of the arc to the center of the circle of which the arc is a part. This section shows how to select a fill pattern and print a filled block or arc. Drawing Filled Blocks The following command sequence prints the block shown below. !R! RES; UNIT P; MZP 72, 72; PAT 6; BLK 72, -144, H; PAGE; EXIT; Figure 2. 8. A Filled Block Lines 1 and 2 put the printing system in the PRESCRIBE mode, reset printing system parameters and set the unit of measurement to points. (One point is equal to 1/72 inches.) Next, the MZP command moves the cursor to the position 72 points to the right of the left edge limit and 72 points below the top edge limit.

The PAT (select fill PATtern) command on line 4 of the program selects the fill pattern. In this program, pattern number 6 is selected. You can select from among any of the printing system's 60 predefined fill patterns or choose to define a pattern using the XPAT (generate eXpanded PATtern) command.

In either case, the selection is made with the PAT command. For 1200-dpi and 600-dpi models, the user can define the printing resolution (300, 600, 1200 dpi) of the pattern by giving a second parameter to the PAT command. You can also select a shade of gray for filling the arc or block by using the GPAT (set Gray PATtern) command. It is possible to apply a color to a pattern specified using PAT, FPAT, GPAT or XPAT. Note, however, that this may not always result in the exact same pattern as printed in monochrome. The way a color looks may differ when used for different patterns even though the same color has been specified. 2-10 The BLK (draw filled-in BLocK) command on line 5 actually draws the filled in block.

This command closely resembles the BOX command explained in the preceding section. However, whereas the BOX command draws a line around a rectangular area, the BLK command fills a rectangular area with the currently selected pattern. As with the BOX command, the position of the rectangular area with respect to the cursor depends on the sign of the values specified for width and height. The box is drawn to the right of the cursor if width is positive, and to the left of the cursor if width is negative; and the box is drawn below the cursor if height is positive, and above the cursor if height is negative. As with the BOX command, you can specify an option parameter to make the cursor move to a specified location after the box is drawn.

(The cursor remains unmoved if the option parameter is omitted.) Values for this option are H, V, E, L, N, and B, the same as for BOX. Drawing Filled Arcs The ARC (draw filled-in ARC) command is similar to the BLK command (described in the preceding section) in that it fills an area with a pre-defined pattern or a shade of gray. The arc is drawn centered around the current cursor position. The dimensions of the arc are determined by user specified inner radius, outer radius, starting angle, and ending angle.

The following PRESCRIBE demonstrates the ARC command. !R! CMNT Enter PRESCRIBE mode; RES; CMNT Reset printing system parameters; UNIT C; CMNT Set centimeters as unit; PAT 9; CMNT Select pattern 9; MZP 8, 8; CMNT Move cursor to point that is 8 cm; CMNT from left edge limit and; CMNT 8 cm from top edge limit; ARC 1, 2, 0, 90; PAGE; EXIT; Figure 2. 9. A Filled Arc The ARC command on line 8 of the command sequence draws an arc with an inner radius of 1 centimeter, an outer radius of 2 centimeters, a starting angle of 0 (straight up), and an ending angle of 90 degrees. The ARC command does not draw a line around the boundary of the filled-in area. 2-11 Chapter 2 Graphics Tutorial Defining Fill Patterns With a little work, you can construct your own fill patterns. You can generate 8 × 8 dot patterns using the FPAT (generate Fill PATtern) command, or 16 × 16 dot patterns using the XPAT (generate eXpanded fill PATtern) command. This section gives examples of both. !R! RES; MZP 1, 1; FPAT 16, 40, 68, 130, 65, 34, 20, 8; BLK 1, 1; PAGE; EXIT; Line 4 of this command sequence prints a filled block using a fill pattern defined by the FPAT command on line 3. Each of the eight numbers in the FPAT defines one row of an 8 × 8 dot pattern.

The pattern follows: Figure 2. 10. Dot Pattern and a Filled Block 128 64 32 16 8 4 2 1 = 16 = 40 = 68 = 130 = 65 = 34 = 20 =8 For this pattern, the numbers across the top indicate the value of each column. The numbers down the right side are the sums of the values of columns that contain black dots in that row. Once this pattern has been defined by the FPAT command, it is used as the fill pattern until printing system parameters are reset with RES, another pattern is selected with PAT, a different pattern is defined with FPAT, or a shade of gray is defined and selected by GPAT. Now let's look at an example using the XPAT command. The XPAT command uses the format XPAT pattern-number; bit map; Note that the pattern-number parameter must be a value from 100 to 105 and followed by a semicolon, not a comma. The following example demonstrates the XPAT command in a PRESCRIBE command sequence. 2-12 !R! RES; XPAT 100; @X0@ | 0Af0CC0FA8L@<X@6p@3p@3X@6L@<FA8CC0Af0@ | 0@X0; MZP 1, 1; PAT 100; BLK 1, 1; PAGE; EXIT; Lines 2 and 3 define the pattern shown in the figure on the next page, defining it as pattern 100. The PAT command on line 5 selects the pattern for use in fills.

Line 6 prints the filled block. Figure 2. 11. Dot Pattern and a Filled Block 16 bits 32 16 8 4 2 1 32 16 8 4 2 1 8 4 2 1 6 bits (x) + 64 6 bits (y) + 64 4 bits (z) + 48 The pattern is 16 dots high and 16 dots wide, and is encoded as a series of 16-bit words. Each 16-bit word is encoded by three characters, representing the most significant six bits, the next six bits, and the least significant four bits, respectively, as shown on the next page.

You obtain the characters that define the pattern by dividing each row-work into sections of six, six, and four bits, calculating the numerical value of each section (referred to as x, y, and z, respectively), treating it as a binary number in which the white dots are zeroes and the black dots are ones. Then add 64 (decimal) to the values of the 6-bit sections and 48 to the values of the 4-bit sections. The result is the ASCII code of the character that represents that section. 2-13 Chapter 2 Graphics Tutorial Column value 0+64=64 (@) 24+64=88 (X) 0+48=48 (0) 0+64=64 (@) 60+64=124 (|) 0+48=48 (0) 1+64=65 (A) 38+64=102 (f) 0+48=48 (0) 3+64=67 (C) 3+64=67 (C) 0+48=48 (0) 6+64=70 (F) 1+64=65 (A) 8+48=56 (8) 12+64=76 (L) 0+64=64 (@) 12+48=60 (<) 24+64=88 (X) 0+64=64 (@) 6+48=54 (6) 48+64=112 (p) 0+64=64 (@) 3+48=51 (3) 48+64=112 (p) 0+64=64 (@) 3+48=51 (3) 24+64=88 (X) 0+64=64 (@) 6+48=54 (6) 12+64=76 (L) 0+64=64 (@) 12+48=60 (<) 6+64=70 (F) 1+64=65 (A) 8+48=56 (8) 3+64=67 (C) 3+64=67 (C) 0+48=48 (0) 1+64=65 (A) 38+64=102 (f) 0+48=48 (0) 0+64=64 (@) 60+64=124 (|) 0+48=48 (0) 0+64=64 (@) 24+64=88 (X) 0+48=48 (0) If the character resulting for section x of any row is @ (indicating that all bits in that section are white), then that character may be omitted. If sections x and y are both @, then both characters may be omitted.

However, if the result for section y is @ and that for section x is a character other than @, then no characters may be omitted. In terms of the program example above, what this means is that the bit map string, @X0@ | 0Af0CC0FA8L@<X@6p@3p@3X@6L@<FA8CC0Af0@ | 0@X0; may be shortened by four characters to: X0 | 0Af0CC0FA8L@<X@6p@3p@3X@6L@<FA8CC0Af0 | 0X0; Patterns defined by the XPAT command remain effective until they are redefined by another XPAT command, or until the printing system is turned off. Drawing Pie Charts The standard graphics mode provides a convenient function for drawing pie charts. See the following example: !R! RES; UNIT C; SPD .05; MZP 10, 10; PIE 2, 0, 10, 20, 30, 40; PAGE; EXIT; 2-14 Figure 2.

12. PIE Example The PIE command uses the format PIE radius, starting angle, size of slice, ...; In the example above, the radius is 2 centimeters (since we set the unit to centimeters with the UNIT command), and the starting angle is 0 degrees.

Four pie slices are specified, with sizes of 10, 20, 30, and 40. The printing system automatically converts the slice sizes to angles totalling 360 degrees. Then it draws the first slice with a cut at the angle specified by the second parameter (0 degrees in our example, or straight up). The remaining slices are drawn in sequence clockwise around the circle. The line thickness used for drawing the circle and the lines between slices are designated by the SPD (Set Pen Diameter) command. Any number of pie slice sizes can be specified, provided that the total length of the command does not exceed 255 characters, and that the sum of the pie slices does not exceed 9999. All numbers specified for slice sizes must be non-negative integers. The PIE command does not fill in the slices with any fill pattern. The PAT command can be used to create shaded areas. The previous example is expanded to fill in the slices.
!R! RES; UNIT C; SPD .1; MZP 10, 10; PAT 19; ARC 0, 2, 0, 36; PAT 41; ARC 0, 2, 36, 108; PAT 43; ARC 0, 2, 108, 216; PAT 48; ARC 0, 2, 216, 360; PIE 2, 0, 10, 20, 30, 40; PAGE; EXIT; 2-15 Chapter 2 Graphics Tutorial Figure 2. 13. Pattern Filled PIE This program first draws four filled arcs, each using a different fill pattern, then prints the pie chart over the arcs. Each arc has an inner radius of zero, an outer radius of 2 (the same as the pie chart), and a starting angle and ending angle that correspond to the relative size of the pie slices.
Since the total size of the pie slices in the example is 100 (10+20+30+40), the angular extent of each arc is equal to 360 x size of slice /100. For example, the angular extent of the first arc is 360 x 10 /100 = 36 degrees. The starting angle of each arc equals the starting angle of the pie chart (0 degrees), plus the angular extent of all the preceding arcs. The ending angle equals the starting angle plus the angular extent of the arc. Path Mode Graphics With path mode graphics, images are constructed by defining lines and curves as paths, then rendering them as images by stroking along the paths or filling the area enclosed by them.
PRESCRIBE provides a variety of path construction operators and painting operators for stroking or filling paths. Path In PRESCRIBE, a path is a set of straight or curved line segments, either connected or disconnected, that describes the shape and position of one or more objects or regions. Paths can be used to draw lines and curves and to specify boundaries of filled areas. A path is stroked by drawing a line of arbitrary width along it. The line may be solid black, all white, or any intermediate shade of gray. It may also be a dashed line of any pattern of segment lengths. A path is filled by painting the entire area that it encloses with a gray scale pattern, ranging from black to white, or with one of the printing system's predefined patterns. In order to be filled, a path must be closed; that is, it must return to its starting point. A path is constructed by means of one or more path construction operators. The path construction operators modify the current path, usually by appending to it.

However, a path in itself does not produce any image on the page. Once a path has been constructed, it can be used to control the application of one of the painting operators of PRESCRIBE, defining the boundary of the area in which images can be printed. 2-16 Path Mode Graphics There are no restrictions on the shape of a path. A single path may include multiple closed subpaths, representing several areas, and a path may intersect itself in an arbitrary manner. The order of the segments that define a path is significant. A pair of line segments is said to connect only if they are defined consecutively, with the second segment starting where the first one ends. Non-consecutive segments that meet or intersect fortuitously are not connected. A subpath is a sequence of connected segments. A path is made up of one or more subpaths. Subpaths may be either open or closed.
Path construction begins with a NEWP (NEW Path) command. Path construction ends with the CLSP (CLoSe Path) command or with any paint operator that paints the region enclosed by the path or draws a line along it (such as STRoKe or FILL). 2-17 Chapter 2 Graphics Tutorial Drawing Lines The following example shows how to draw a line in the path mode. !R! RES; NEWP; PMZP 1, 1; PDZP 2, 3; STRK; PAGE; EXIT; Figure 2. 14.
Drawing Lines in Path Mode (1, 1) (2, 3) Line 1 of the program switches the printing system to the PRESCRIBE mode and resets printing system parameters, including the unit (to inches), line width (to 3 dots), and various other aspects of the graphics state. Path construction begins with the NEWP command on line 2. This command empties the current path (if any), making it possible to start a new one. In doing so, it makes the position of the cursor undefined. The PMZP (Path, Move to Zero-relative Position) command on line 3 moves the cursor to a position one inch from the top and left edge limits of the paper.
The coordinates specified may be positive or negative. On line 4, the PDZP (Path, Draw to Zero-relative Position) draws a line from the current cursor position to the position 2 inches from the left edge limit and 3 inches from the top edge limit. The cursor remains at this position after the line is drawn. On line 5, the STRK command strokes the path onto the page. After stroking the current path, the STRK command clears the path in the same manner as NEWP (start NEW Path). Finally, PAGE prints out the page, allowing us to look at the result of our work and cancelling all changes made during the course of the program, then EXIT ends the PRESCRIBE mode. 2-18 Path Mode Graphics Two Lines The preceding example illustrated construction of a path between points specified in terms of absolute coordinates. The following program draws two lines, using both absolute coordinate specification and a new method: relative coordinate specification. !R! RES; NEWP; PMZP 1, 1; PDZP 2, 3; PMRP .5, -1; PDRP -1, -1; SPD 0.

04; STRK; PAGE; EXIT; Figure 2. 15. Drawing Two Lines (1, 1) (-1, -1) (0.5, -1) (2, 3) The first four lines of this program are identical to the preceding example. Line 1 switches the printing system to the PRESCRIBE mode and resets printing system parameters, line 2 empties the current path, and lines 3 and 4 draw a line between two points that are specified in terms of absolute coordinates. On line 5, the PMRP (Path, Move to Relative Position) command moves the cursor to the point half an inch to the right and one inch above the current cursor position; that is, the point at which the first line ends. Then the PDRP (Path, Draw to Relative Position) command on line 6 draws a line to the point 1 inch to the left of the new position and 1 inch below it.

The line thickness is changed to 0.04 inches by the SPD command on line 7. Finally, the STRK command on line 8 strokes the path onto the page, PAGE prints out the page, and EXIT ends the PRESCRIBE mode.

2-19 Chapter 2 Graphics Tutorial Line Ends The line end type determines how PRESCRIBE renders the ends of lines when they are stroked onto the page. PRESCRIBE provides three kinds of line ends. These include: Figure 2. 16. Line Ends Square caps Butt caps Round caps The default line end type is butt caps.

You can switch from the current line end type to any of the other types with the SCAP command. This command uses the following format: SCAP line-cap mode; Values for line-cap mode include: 1 (for square caps) 2 (for butt caps) 3 (for round caps) Use of this command is illustrated in the following example. !R! RES; UNIT C; CMNT Sets unit to cm; NEWP; CMNT Starts new path; SPD .5; CMNT Sets line width to .5 cm; SCAP 1; CMNT Sets square caps; PMZP 2, 2; PDZP 4, 4; SCAP 3; CMNT Sets round caps; STRK; PAGE; EXIT; Figure 2.

17. Printout of SCAP Example Note that the line is rendered with round caps, rather than with square ones. Although square caps is set before constructing the path, the line type is changed to round prior to 2-20 Path Mode Graphics stroking the path. PRESCRIBE refers to the line cap type when the current path is stroked onto the page, rather than while the path is being constructed. Therefore, the program above renders the line with round caps rather than square ones. Line Joins When a path consists of multiple connected line segments, the manner in which they are stroked onto the page depends on the current line join type. PRESCRIBE provides four types of line joins. These are called beveled, mitered, round, and notched. These are illustrated below. Figure 2.

18. Joins Mitered join Round join Notched join Beveled join The default line join type is beveled. With beveled joins, connected line segments end with butt caps, and the notch at the larger angle between the segments is filled with a triangle. With mitered joins, the edges of connected line segments are extended until they meet. This type of join is limited by the miter limit (explained below). With round joins, connected line segments are joined with circular caps. Notched joins leave a notch at the larger angle between the connected line segments. You can switch from the current line join type to any of the other types with the SLJN (Set Line JoiN) command. This command uses the following format: SLJN line-join mode; Values for line-join mode include: 1 (for beveled joins) 2 (for mitered joins) 3 (for round joins) 4 (for notched joins) Miter Limit When using mitered line joins, the use of such joins is limited by the miter limit. The miter limit is the maximum ratio of the distance l between the inner and outer corners of a mitered join and the width w of the lines joined.

2-21 Chapter 2 Graphics Tutorial Figure 2. 19. Miter Limit W W = line width L = miter length L Miter limit = maximum ratio of L/w = 1/sin (a/2) If the angle at which lines join is such that this limit is exceeded, the lines are joined with a beveled join, rather than a mitered one. The purpose of the miter limit is to prevent objectionably long spikes when lines join at small angles. The default miter limit is 10, which results in beveled joins at angles of less than about 11. 5 degrees. You can set any desired miter limit with the SMLT (Set Miter LimiT) command. This command has the following format. SMLT limit-value; Here are some representative limit-values and the corresponding angles at which the line join type switches between mitered and beveled. limit-value 2 3 4 5 6 7 8 9 appox.

angle 60 39 29 23 19 16 14 13 Dash Type By default, the STRK command strokes paths with solid lines. However, you can also use a predefined pattern of alternating black and white to stroke paths. This makes it possible to stroke paths as dashed lines. You can also define your own dashed line patterns. The DPAT (select Dash PATtern) command selects one of PRESCRIBE's ten predefined dash patterns, or one of 10 dash patterns that you can define yourself. This command uses the format: DPAT pattern-number; 2-22 Path Mode Graphics The following program illustrates use of this command. !R! RES; UNIT C; CMNT Sets unit to cm; NEWP; CMNT Starts new path; SPD .5; CMNT Sets line width to .5 cm; PMZP 2, 2; PDZP 4, 4; DPAT 5; STRK; PAGE; EXIT; Figure 2. 20.

Printout of the DPAT Example In this program, the DPAT command selects the dash pattern with which the line is stroked. Predefined dash patterns are selected by specifying values from 1 to 10 for pattern-number. (A value of 1 specifies solid lines.) User-defined patterns can be selected by specifying values from 11 to 20. The next section explains how to use the SDP (Store Dash Pattern) command to define your own dash patterns. Specifying an undefined user pattern number results in solid black lines. User Defined Dash Patterns Using the SDP command, you can define your own dashed patterns for use in stroking lines, arcs, and curves. See the following example: !R! RES; UNIT P; SDP 11, 2, 2, 5, 2; UNIT C; CMNT Sets unit to cm; NEWP; CMNT Starts new path; SPD .5; CMNT Sets line width to .5 cm; PMZP 2, 2; PDZP 4, 4; DPAT 11; STRK; PAGE; EXIT; 2-23 Chapter 2 Graphics Tutorial Figure 2.

21. Printout of the SDP Example The SDP command on line 3 defines a dashed pattern consisting of two lengths of black, two lengths of white, five lengths of black, and two lengths of white. On line 11, the DPAT command selects this pattern for stroking. The path defined by the PMZP (Path, Move to Zero-relative Position) and PDZP (Path, Draw to Zero-relative Position) commands is stroked using this pattern, with a result as shown in the figure above. The SDP command uses the general format: SDP pattern-number, dash1, space1, dash2, space2, .

.., dash10, space10; The value specified for pattern-number must be in the range from 11 to 20. Dash and space lengths are specified in pairs. Up to ten dash-space pairs can be specified.

The dash length always comes first in each pair. If a pattern is to begin with a space, then specify 0 for dash1. However, if a pattern ends with a dash, the following space parameter can be omitted. Dash patterns defined with this command remain valid until redefined with another SDP command, or until the printing system is turned off. Drawing Arcs and Curves A path can include curves as well as lines. The PARC (Path, draw ARC) and PCRP (Path, Curve to Relative Position) commands make it possible to draw circular arcs and arcs of more complex form. The PARC command uses the format: PARC x, y, radius, ang1, ang2; where x and y describe the zero-relative coordinates of the center of the arc, radius describes the radius of the arc, ang1 describes the arc's starting angle, and ang2 describes the arc's ending angle. Coordinates and radius are measured in the unit currently designated by the UNIT command, and the starting and ending angles are measured clockwise from the positive x axis.