10 User-Defined Symbol Sets

Introduction

User-defined symbol sets are used with unbound scalable fonts. Three new commands provide for the implementation of user-defined symbol sets:

Symbol Set ID Code - $^{E}_{C}$***c#R**

Define Symbol Set - $^{E}_{C}$**(f#W [symbol set data]**

Symbol Set Control - $^{E}_{C}$***c#S**

To define a symbol set, you must first designate a symbol set ID code. Next, use the Define Symbol Set command to download the list of characters (character codes and related symbol index numbers) for the symbol set. Once this is accomplished, you may select the symbol set for printing in the same manner as any symbol set using the symbol set selection sequence: $^{E}_{C}$**(ID**

Once a user-defined symbol set is downloaded, the Symbol Set Control command can be used to assign symbol sets as either temporary or permanent and to delete them.

Symbol Set ID Code Command

The Symbol Set ID Code command assigns a symbol set ID code to a user-defined symbol set. This ID code is used by the Symbol Set Control command for symbol set management.

The ID code corresponds to the symbol set ID selection value which is used to identify the symbol set during font selection.

E_C *** c # R**

\# = Symbol Set ID Code (decimal)

Default = 0

Range = 0 - 32,767 (larger values are outside the legal range)

When downloading a symbol set, the symbol set ID value must match the Encoded Symbol Set Designator field in the user-defined symbol set header.

As mentioned above, the symbol set ID code is related to a symbol set ID value. The relationship between the ID code and the symbol set ID selection value is shown by the following formula:

Symbol Set ID code = (# * 32) + (ID - 64)

where: "#" represents the number portion of the ID selection value which may range from 0-1023; and, "ID" represents the ordinal (decimal) value of the ID character. (Symbol set ID selection values consist of a number and a letter, such as 8U for Roman-8 or 7J for DeskTop etc.)

For example:

Assume the ID selection value selected for this symbol set is 17Q, then:

(17 * 32) + (81 - 64) = 561

The symbol set ID code is 561.

When selecting an ID code, select one which is not being used currently. (If an ID code is selected which is already being used in the printer, that symbol set is redefined.) The first step in selecting an ID code is to determine an ID selection value. Since it may be difficult to determine which ID selection values are being used, it is best to select one which has not been assigned. Refer to Appendix C in the *PCL 5 Comparison Guide* to select a number/letter (ID selection value) combination which has not been assigned.

After the ID selection value has been determined, use the conversion formula (shown above) to convert the ID value to an ID code.

To create a user-defined symbol set:

1 Identify the symbols (symbol index numbers) for the symbol set from the MSL or Unicode list in Appendix D of the *PCL 5 Comparison Guide*. List them in the appropriate character code order.

2 Identify the symbol collections (and Character Requirement bits) that contain the symbols (Appendix D of the *PCL 5 Comparison Guide*).

3 Identify the Define Symbol Set command header information.

4 Identify a symbol set ID selection value and convert it into its symbol set ID code.

5 Designate the ID code using the Symbol Set ID Code command.

6 Download the symbol set header data and MSL or Unicode numbers using the Define Symbol Set command.

To print using the symbol set, it must be selected using the Select Symbol Set command - $^{E}{}_{C}$**(ID**, where **ID** is the symbol set ID selection value used to calculate the symbol set ID code.

Define Symbol Set

This command defines the characters and character mapping for a user-defined symbol set.

$^{E}_{C}$ (f # W [symbol set definition data]

= Number of bytes in symbol set definition

Default = n/a
Range = 0 - 32767

If a user-defined symbol set is already present in the printer and a new, valid symbol set is downloaded with the same Symbol Set ID Code, then the old symbol set is deleted. If an internal symbol set with the same ID code exists, it is overridden by the new symbol set. If the symbol set definition is invalid, the command is ignored. If there is insufficient memory to create the symbol set, the symbol set is discarded.

Note

Send the Symbol Set ID Code command prior to the Define Symbol Set command to assign an ID code for the user-defined symbol set. If the Symbol Set ID Code command is not sent, the last code sent is used. If none have been sent, then the default (0) is assigned.

The data format for the user-defined symbol set is shown in Table 10-1.

Table 10-1 User-Defined Symbol Set Defintion Format

Byte	15 - MSB8	7LSB - 0
0	Header Size (18)	
2	Encoded Symbol Set Designator	
4	Format	Symbol Set Type
6	First Code	
8	Last Code	
10	Character Requirements	
Hdr Size	Symbol Map [Last Code — First Code + 1] ⋮	

The following abbreviations are used to define the data type of each field in the symbol set header:

Font Header Field Data Type Notation		
(B)	: Boolean	(0, 1)
(UB)	: Unsigned Byte	(0 . . 255)
(SB)	: Signed Byte	(-128 . . 127)
(UI)	: Unsigned Integer	(0 . . 65535)
(SI)	: Signed Integer	(-32768 . . 32767)
(ULI)	: Unsigned Long Integer	(0 . . 2^{32}-1)
(SLI)	: Signed Long Integer	(-2^{31} . . 2^{31}-1)
(ASCxx)	: ASCII string	array (0 . . xx-1) of characters

Header Size (UI)

Set the header size to the size of the header data — the number of bytes from Header Size (byte 0) to the last byte just before the beginning of the Symbol Map data bytes. This value is 18 or greater.

Encoded Symbol Set Designator (UI)

This field must match the ID code in the Symbol Set ID code command.

This field contains the symbol set ID code. The symbol set ID code is calculated from a symbol set ID selection value using the following formula:

Symbol Set ID Code = (# * 32) + (ID - 64)

where # is the decimal number (0 to 1023) and ID is the ASCII character code of the letter.

For example:

Assume the ID selection value selected for this symbol set is 17Q, then:

(17 * 32) + (81 - 64) = 561

The symbol set ID code is 561.

Format (UB)

Set this field to 1 for MSL (Intellifont) or 3 for Unicode (TrueType).

Unrecognized values cause the symbol set definition to be ignored.

Symbol Set Type (UB)

This field defines the printable and unprintable codes for the symbol set.

Bit Field	Designated Use
0	7-bit, 32-127 are printable.
1	8-bit, 32-127 and 160-255 are printable.
2	8-bit, 0 - 255 character codes are printable, however, to print codes 0, 7-15, and 27, the printer must be in transparency mode.

First Code (UI)

Specifies the first character code in the set.

Last Code (UI)

The Last Code specifies the last character code in the set.

Together, the First Code through the Last Code identify the range of character codes which map to the symbol index numbers (characters) in the Symbol Map field.

Character Requirements (Array of UB) and character requirement

This 8-byte field works in conjunction with the Character Complement field in the header of a type 10 or 11 (unbound) font to determine the compatibility of a symbol set with an unbound font. These two fields identify the unbound fonts in the printer which contain the symbol collections required to build a symbol set. Refer to “Unbound Scalable Fonts” in Chapter 9, for a description of symbol collections and unbound fonts.

Each bit in the field represents a specific collection. Setting a bit to 1 indicates that collection is required; setting the bit to 0 indicates that collection is not required. (Bit 63 refers to the most significant bit of the first byte, and bit 0 refers to the least significant bit of the eight byte field.) The bit representations for the collections are shown below. (The symbols for each collection are shown in Appendix D of the *PCL 5 Comparison Guide*.)

MSL Symbol Index Character Requirements		
Bit	**Value**	**Designated Use**
63	1	Basic Latin required (such as ISO 8859/1 Latin 1)
	0	Basic Latin not required
62	1	East European Latin required (such as ISO 8859/2 Latin 2).
	0	East European Latin not required
61	1	Turkish required (such as ISO 8859/9 Latin 5)
	0	Turkish not required
34	1	Math required (such as Math-8)
	0	Math not required
33	1	Semi-graphic required (such as PC-8 D/N)
	0	Semi-graphic not required
32	1	Dingbats required (such as ITC Zapf Dingbats series 100, series 200, etc.)
	0	Dingbats not required
2,1,0	000	MSL Symbol Index

Unicode Symbol Index Character Requirements Bits (numbers/values)		
Bit	**Value**	**Designated Use**
31	1	ASCII required (such as ISO 6 ASCII)
	0	ASCII not required.
30	1	West Europe extensions required (such as ISO 69 French).
	0	West Europe extensions not required.
29	1	East Europe extensions required (such as ISO 8859/2 Latin 2).
	0	East Europe extensions not required.
28	1	Turkish extensions required (such as ISO 8859/9 Latin 5).
	0	Turkish extensions not required.
27	1	Desktop Publishing extensions required (such as Windows 3.1).
	0	Desktop Publishing extensions not required.
26	1	Accent extensions required (such as ISO 8859/1 Latin 1).
	0	Accent extensions not required.
25	1	PCL extensions required (such as Roman-8).
	0	PCL extensions not required.
24	1	Macintosh extensions required (such as MC Text).
	0	Macintosh extensions not required.
23	1	PostScript extensions required (such as PS Text).
	0	PostScript extensions not required.
22	1	Code Page extensions required (such as PC-8).
	0	Code Page extensions not required.
2,1,0	001	Unicode Symbol Index

Examples of values for the field include:

Bit Field	Designated Use
Value (Hex)	**Meaning**
0000000000000000	Default requirement (MSL); symbol set can be used with any typeface indexed bu MSL.
8000000000000000	Symbol set (MSL) requires only the Basic Latin Symbol Collection (such as Roman-8)
0000000100000000	Symbol set (MSL) requires only the Dingbat Collection.
0000000000000001	Default requirement (Unicode); symbol set can be used with any typeface indexed bu Unicode.
00000000A0000001	Symbol set (Unicode) requires the ASCII and East Europe Collections (such as ISO 8859/2).
0000000088000001	Symbol set (Unicode) requires the ASCII and Desktop Publishing Collections (such as Ventura US).

Symbol Map (Array of UI)

The symbol map contains a list of symbol index numbers. This list identifies symbols for the symbol set. (Refer to Appendix D in the *PCL 5 Comparison Guide* for a MSL and Unicode symbol indexes.) The symbol map pairs (maps) a character code to a symbol index number. The range of character code numbers (paired with symbol index numbers) is the range from the First Code through the Last Code fields in the header. The first symbol index number in the Symbol Map field is mapped to the character code whose value is that of the First Code field; the second symbol index number is mapped to the "First Code + 1" character code; the third symbol index number is mapped to the "First Code + 2," etc., through the last symbol index number, which is mapped to the value in the Last Code field. The number of symbol index characters in the array must match the number of character codes in the range, First Code through Last Code.

If no printable symbol (symbol index number) is associated with a given character code (as with codes 128 through 160 of Roman-8), the corresponding entry in the Symbol Map should be 65535 (FFFF Hex).

Symbol Set Control Command

This command provides a means for making user-defined symbol sets permanent or temporary, and for deleting them.

E_C *** c # S**

= 0 -Delete all temporary and permanent user-defined symbol sets.
1 - Delete all temporary user-defined symbol sets.
2 - Delete current user-defined symbol set (last symbol set ID code specified).
4 - Make current user-defined symbol set temporary.
5 - Make current user-defined symbol set permanent.

Default = n/a
Range = 0-2, 4, 5 (other values ignored)

Downloaded symbol sets default to temporary.

Internal symbol sets cannot be deleted or made temporary.

User-Defined Symbol Set Examples

The following two examples illustrate the concept of user-defined symbol sets. They create symbol sets for PC-8 in MSL and Unicode symbol indexes. The necessary escape sequences are shown in each example.

Unicode Symbol Index Example

Symbol Map Data:	
$^{E}_{C}$*c341R	PCL Symbol Set #IDs: 10U
$^{E}_{C}$(f526W	Symbol Set 526 bytes in length
00 12	Header Size 18 bytes
01 55	ID code 341 decimal: 10U
03	Format 3 (Unicode Symbol Index)
02	Font Type 2
00 01	First code = 1
00 fe	Last code = 254
00 00 00 00 c0 40 00 01	ASCII, Latin 1, and PC Characters required
26 3a (character code 1)	Open Happy Face
26 3b (character code 2)	Solid Happy Face
26 65 (character code 3)	Solid Heart, Card Suit
26 40 (character code 4)	Solid Diamond, Card Suit
26 63 (character code 5)	Solid Spade, Card Suit
⋮	⋮
25 bc (character code 31	Down Solid Arrowhead
ff ff (character code 32)	Space Code (no character)
00 21 (character code 33)	Exclamation Mark
⋮	⋮

Symbol Map Data: ***(continued)***	
00 41 (character code 65)	Uppercase A
⋮	⋮
00 61 (character code 97)	Lowercase A
⋮	⋮
20 7f (character code 252)	Superior Lowercase N
00 b2 (character code 253)	Superior Numeral 2
25 a0 (character code 254)	Small Solid Square Box
$^{E}_{C}$*c341r5S	PCL Symbol Set #ID: 10U, Make this symbol set permanent.

MSL Symbol Index Example

Symbol Map Data:	
$^{E}_{C}$*c341R	PCL Symbol Set #ID: 10U
$^{E}_{C}$(f528W	Symbol Set 528 bytes in length
00 12	Header Size 18 bytes
01 55	ID code 341 decimal: 10U
01	Format 1 (MSL Symbol Index)
02	Font Type 2
00 01	First code = 1
00 ff	Last code = 255
80 00 00 02 00 00 00 00	Basic Latin and PC Characters required
00 cb (character code 1)	Open Happy Face
00 cc (character code 2)	Solid Happy Face
00 cd (character code 3)	Solid Heart, Card Suit
00 ce (character code 4)	Solid Diamond, Card Suit

Symbol Map Data: ***(continued)***	
00 cf (character code 5)	Solid Spade, Card Suit
⋮	⋮
00 e7 (character code 31)	Down Solid Arrowhead
00 00 (character code 32)	Space Code
00 01 (character code 33)	Exclamation Mark
⋮	⋮
00 22 (character code 65)	Uppercase A
⋮	⋮
00 43 (character code 97)	Lowercase A
⋮	⋮
01 4c (character code 252)	Superior Lowercase N
00 c5 (character code 253)	Superior Numeral 2
01 31 (character code 254)	Small Solid Square Box
00 00 (character code 255)	No-Break Space
$^{E}{}_{C}$*c341r5S	PCL Symbol Set #ID: 10U,