ENHANCING THE PRINTED APPEARANCE OF YOUR SAS® REPORTS

S. David Riba, JADE Tech, Inc.

INTRODUCTION

Are you tired of having all of your SAS® reports print in Lineprinter or Courier 10 pitch? Have you ever wanted to use some of the advanced features of your printer, but were uncertain how to access them? For the Laser Printer user, wouldn’t it be useful to select Landscape or Portrait mode for different reports within the same job? Or to selectively change the font or features such as Bolding or Italic within a report? If you are a dedicated Lotus 1-2-3® user, you probably already know how to turn on CompressedPrint by sending \015 to the printer, or Expanded Print with \02710871001, and so on. Wouldn’t it be useful to do the same with your SAS reports?

The SAS System has limited built-in capabilities to customize the appearance of finished output. Most SAS users typically use the system defaults when printing output, even when the printers they use for their SAS reports have features that could be used to enhance the appearance of the final SAS output. If the printer features are used at all, they are typically accessed by manually changing the control panel options on the printer prior to printing. This allows the use of global features, such as font or print size, which become effective for an entire print job.

There is a way to control specific printer features in SAS, and it is not too difficult. While the examples below specifically discuss printing features of Dot Matrix and Laser Printers, these techniques should also work on any printer with user controllable features.

CONTROLLING YOUR PRINTER

The key to solving this problem is the Escape Code Table used by your printer. Escape (or Control) Codes are usually ASCII code sequences. They may be represented as a Decimal number, a Hexadecimal number, or a Control Key combination. For example, the code for a Form Feed is “12” (Decimal), “0C” (Hexadecimal), or “CTRL L” (Control Key and the letter L). When a printer receives any of those sequences, it converts the sequence to Binary format and interprets the sequence as a Form Feed instruction. There is usually a section in the back of most printer manuals with the usable features of that printer and their associated Escape Codes. In addition, some manuals also provide the Escape codes in both Decimal and Hexadecimal. The codes that follow should work properly for most printers, if in doubt check your manual.

Below are some typical Printer Control Codes that you might see in a printer manual:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Code (Decimal)</th>
<th>Code (Hexadecimal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portrait Printing</td>
<td>ESC&amp;IOO</td>
<td>1B266440</td>
</tr>
<tr>
<td>Landscape Printing</td>
<td>ESC&amp;10</td>
<td>0C266440</td>
</tr>
<tr>
<td>Italic Print</td>
<td>ESC(a1S)</td>
<td>61266440</td>
</tr>
<tr>
<td>Double Wide Print</td>
<td>ESC 50</td>
<td>30266440</td>
</tr>
</tbody>
</table>

If your printer manual does not provide the Hex values associated with the ESCape sequences, use a conversion chart to translate the codes from ESCape sequence to Decimal or Hexadecimal values. For example, the ESCape key can be coded as:

```
^L  Control Key and Left Bracket Key
"7"  Decimal
"4B"  Hexadecimal
```

Convert each item in the sequence to build a Hexadecimal Code string. For example, to change ESC&10 (Landscape Print) to Hexadecimal:

```
ESC & IOO
\ is hex 1B
& is hex 26
I (lower case L) is hex 6C
1 (number one) is hex 31
O (upper case O) is hex 4F
```

The completed hex code for Landscape Print is 1B266C314F (note that there are NO spaces in the completed hex code).

USING CONTROL CODES IN SAS PROGRAMS

The simplest way to select a feature on your printer is to “print” the Hex Code for that feature using PUT statements. For example, to set the page orientation to Portrait mode on most Laser Printers, the Hex Code is “1B266C304F” (Hex for ESC&10). To change to Portrait mode, your SAS Program could include:

```
DATA _NULL_ ;
FILE "LPT1:" ;
PUT "1B266C304Fx" ;
RUN ;
```

This will print the character constant “1B266C304F” as a hex string to the printer. When the printer receives this string, it will interpret the string as an ESCape sequence and process it as a command. Remember, there is a difference between 0 (the number zero) and O (capital letter ‘O’) and between 1 (the number one) and l (small letter ‘L’). Use the numbers and NOT the letters in the Hex Codes above.

Once enabled, some features on a printer will continue until disabled. A good example is underlining. Once the printer receives the command to start underlining, it will continue until a command is received to discontinue underlining. In these instances, send a pair of ESCape sequences to control where the feature is to begin and where the feature is to end. In the case of underlining, your code might look something like:

```
DATA _NULL_ ;
FILE "LPT1:" ;
PUT "1B266C444x" ;
RUN ;
```

This is Underlined’1B266440’x

' This is no longer Underlined’ ;
If you use a Dot Matrix Printer, a report writer step might have the following:

```
PUT @10 'OE'x 'THIS IS DOUBLE WIDE' '14'x ;
I
@10 '1B45'x 'THIS IS BOLDED' '1B46'x ;
@10 '1B2D01'x 'THIS IS UNDERLINED' '1B2DOO'x ;
OF 'THIS IS COMPRESSED PRINT' '1 BOF'x ;
```

Not too difficult, once you see how it works. The reason this works is that the SAS System considers hexadecimal numbers the same as any other number. Both HEXw. and $HEXw. are valid SAS hexadecimal informats and formats. In addition, the single character "X" following a number or string identifies a hexadecimal constant. Thus, 'E2C1E2'X is a hex character constant and 0C1X is a hex numeric constant.

- **Character hexadecimal constants**
  - must be enclosed in single quotation marks
  - must be followed immediately by an X
  - can include commas to make the hex string more readable, provided the commas separate an even number of hex characters ('F1,F2,F3F4'X)

- **Numeric hexadecimal constants**
  - start with a number digit (usually zero)
  - are not bounded by quotation marks
  - can be any number of characters
  - must be followed by an X

There are several alternative methods in SAS to embed printer Control Codes in your SAS job. The first method is to type the codes in your PUT statements. However, it is tedious entering each of those ESCape codes every time you want to use them. In addition, every time you type an Escape Code, you introduce the risk of mistyping the code. To prevent "typos" and make your final SAS code much more readable you can

- assign a SAS variable with the value of the ESCape Code.
- create a permanent Format of the Escape Codes.
- use global Macro variables.

### USING SAS VARIABLES

One problem with typing the hex codes directly in a PUT statement is that it takes a sharp eye to detect a minor change that could be the difference between a valid hex code and one that the printer does not recognize. For example, there is only one character difference between '1B266C304F'x (Portrait mode) and '1B266C314F'x (Landscape mode), yet the end result is quite different.

Instead of typing the hex code each time it is needed, consider assigning the code to a SAS variable. Then your SAS code could include statements like:

```
bold = '1B28733142'x ; unbold = '1B28733042'x ;
PUT bold 'This is Bolded' unbold ;
```

### USING GLOBAL MACRO VARIABLES

The major disadvantage of this method is the additional overhead of creating and tracking a SAS variable for each printer Control Code. A more efficient method would be to create a global SAS macro variable for each Control Code. As you develop your program, you could add a table of macro variables like this:

```
%LET DUBLWIDE = 'OE'x;
%LET UNDUBL = '14'x;
%LET BOLD = '1B45'x;
%LET UNBOLD = '1B46'x;
%LET UNDERLIN = '1B2D01'x;
%LET NOUNDLIN = '1B2D00'x;
%LET COMPRESS = '0F'x;
%LET NOCOMPRESS = '1BOF'x;
%LET PORTRAIT = '1B266C304F'x;
%LET LANDSCAP = '1B266C314F'x;
%LET FORMFEED = 'OF'x;
```

Now, you could code your report writer something like this:

```
PUT@10&DUBLWIDE'THISISDOUBLEWIDE'&UNDUBL
@10 &BOLD 'THIS IS BOLDED' &UNBOLD
@10&UNDERLIN 'UNDERLINED' &NOUNDLIN;
PUT &FORMFEED;
```

Not only does this eliminate the possibility of mistyping the printer Control Codes, but it makes your final SAS program much more readable. These macro variables would be available anywhere in your SAS job and would not add additional variables to your SAS data sets.

### USING PERMANENT FORMATS

The most efficient method would be to create a permanent SAS format of the printer Control Codes. For example, if a permanent format called PTR_CODE was created:

```
PROC FORMAT LIBRARY=LIBRARY :
VALUE PTR_CODE 1 = '1B266C304F'X
2 = '1B266C314F'X
3 = '1B266C344'X
4 = '1B266C314F'X
... ...
```

Your report writer might then include code like:

```
undrlin = 3 ; undroff = 4 ;
PUT undrlin ptr_code. 'This is Underlined' undroff ptr_code. ;
```

### CAVEATS ABOUT USING CONTROL CODES

When using printer Control Codes in your SAS reports, be very careful about cursor movement commands to move the column pointer. The use of Control Codes in a report makes the SAS position counter a relative position rather than an absolute position. Since SAS "prints" the Code, the SAS position pointer is moved the number of spaces occupied by the Control Code. However, your printer interprets the Code as an instruction and acts accordingly. Thus, all subsequent printing on a line will start at the position previously occupied by the Control Code. Everything you print after a Control Code will print in a position relative to that Control Code rather than in an absolute position.
Consider the following header:

PUT @40 'Underlined Text' @60 'Next Text';

The following would underscore the first column.

PUT @40 '1B266440'x 'Underlined Text' @60 'Next Text';

However, since the second block of text would start printing at column 60, it would actually overwrite part of the second Control Code. In this instance, underlining would be started at column 40 but the Control Code to stop underlining would never be seen by the printer.

Instead, try:

PUT @40 '1B266440'x 'Underlined Text' @60 'Next Text';

CONCLUSION

Printer Control Codes can easily be included within your SAS programs to allow access to a variety of printer features that were previously not controllable from within SAS. Features such as page orientation, fonts, text bolding, and the ability to print special characters can all be controlled through the use of printer Control Codes. These codes can be "printed" with PUT statements. They can also be stored as either macro variables or as permanent SAS formats for use anywhere in your SAS programs.

---

SAMPLE PRINTER CONTROL CODES

<table>
<thead>
<tr>
<th>Font Style</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>COURIER FONT</td>
<td>1B28733354</td>
</tr>
<tr>
<td>HELVETICA FONT</td>
<td>1B28733454</td>
</tr>
<tr>
<td>12 PITCH PRINT</td>
<td>1B28663453</td>
</tr>
<tr>
<td>8 LINES PER INCH</td>
<td>1B266C3644</td>
</tr>
<tr>
<td>8 LINES PER INCH</td>
<td>1B266C3844</td>
</tr>
<tr>
<td>ITALIC PRINT</td>
<td>1B28733153</td>
</tr>
<tr>
<td>SUPERSCRIPT</td>
<td>1B2A702D320359</td>
</tr>
<tr>
<td>SUBSCRIPT</td>
<td>1B2A702B320359</td>
</tr>
<tr>
<td>TRADEMARK</td>
<td>1B283132551B2670A91B268313055</td>
</tr>
</tbody>
</table>

PUT @40 '1B266440'x 'Underlined Text' @60 'Next Text';

The author may be contacted at:

JADE Tech, Inc.
P. O. Box 4517
Clearwater, FL 34618
(813) 726-6099

SAS is a registered trademark of SAS Institute, Inc., Cary, NC, USA

Lotus 1-2-3 is a registered trademark of Lotus Development Corp., Cambridge, MA, USA

---

ASCII Codes

<table>
<thead>
<tr>
<th>Dec</th>
<th>Hex</th>
<th>Char</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>01</td>
<td>NUL</td>
</tr>
<tr>
<td>2</td>
<td>02</td>
<td>SOH</td>
</tr>
<tr>
<td>3</td>
<td>03</td>
<td>STX</td>
</tr>
<tr>
<td>4</td>
<td>04</td>
<td>ETX</td>
</tr>
<tr>
<td>5</td>
<td>05</td>
<td>EOT</td>
</tr>
<tr>
<td>6</td>
<td>06</td>
<td>ACK</td>
</tr>
<tr>
<td>7</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>8</td>
<td>08</td>
<td>BS</td>
</tr>
<tr>
<td>9</td>
<td>09</td>
<td>HT</td>
</tr>
<tr>
<td>A</td>
<td>10</td>
<td>LF</td>
</tr>
<tr>
<td>B</td>
<td>11</td>
<td>VT</td>
</tr>
<tr>
<td>C</td>
<td>12</td>
<td>FF</td>
</tr>
<tr>
<td>D</td>
<td>13</td>
<td>CR</td>
</tr>
<tr>
<td>E</td>
<td>14</td>
<td>SO</td>
</tr>
<tr>
<td>F</td>
<td>15</td>
<td>SI</td>
</tr>
</tbody>
</table>

---

The author may be contacted at:

JADE Tech, Inc.
P. O. Box 4517
Clearwater, FL 34618
(813) 726-6099

SAS is a registered trademark of SAS Institute, Inc., Cary, NC, USA

Lotus 1-2-3 is a registered trademark of Lotus Development Corp., Cambridge, MA, USA

---

567