

*Just for JGT  
+ PA in his box  
BJR  
Feb. 12/  
85*

PRINTING SCIENTIFIC REPORTS  
ON THE QANTEX PRINTER

by

Richard Bastien

Internal Report 85-1

Division of Gravity, Geothermics and Geodynamics,  
Earth Physics Branch,  
Department of Energy Mines and Resources,  
1 Observatory Cresc.,  
Ottawa, Canada,  
K1A 0Y3.

This document was produced  
by scanning the original publication.

Ce document est le produit d'une  
numérisation par balayage  
de la publication originale.

## ABSTRACT

SCR prepares scientific documents to be printed on the Qantex Model 7030. Printing mode selection responses are for main PROM 884804, revision F and higher. The program is written in FORTRAN-77 and runs under RSX-11M.

It operates on an input file in which columns N+1 to N+5 contains special characters to control the Qantex options that will be applied to the current line. N is specified by the user. The program recognizes 12 special characters which permit: printing in scientific and/or italic format, using the centerline and/or underline option, overprinting or 1/12 inch linefeed. The program provides full control of the printer speed for draft and/or letter quality printing, a choice of different horizontal and vertical pitches and an option for page numbering. The limits are set to 123 columns and 115 lines per page.

Command `>SCR` prepares scientific documents to be printed on the Qantex Model 7030. Printing mode selection responses are for Main PROM 884804, Rev F and higher. The program is written in FORTRAN-77.

It operates on an input file in which columns N+1 to N+5 contains special characters to control the Qantex options that will be applied on the current line. N is specified by the user in the first line of the input file. The program recognizes 12 special characters and the position they hold within columns N+1 to N+5. The maximum length of a text line is 123 characters. TABs used in the input file must have the standard TAB length of 8 columns.

To produce a copy of a document :

`>SCR` and enter the input file name when asked.

The user has an option for page numbering and a menu presents 6 possible options for the number of lines per page. The program will produce a file with the same name as the input file, but with the suffix .QTX . This file can then be sent during runtime to the Qantex if desired or via the command `>PRINT` which contains the remaining options not included in `>SCR`. A line counter will generate a form feed after the optional number of lines per page. No form feed will be generated within a multiline formula. The line counter is reset to zero if a form feed is located in the first column of the input file.

#### Short user's guide

Symbol	Column	Function
'~' (tilde).....N+1	=>	Italic characters option
'.' (period).....N+1	=>	Scientific characters option
'*' (asterisk).....N+2	=>	Centre line option
'_' (underscore)....N+3	=>	Underline option
'+' (plus sign)....N+4	=>	Overprinting
',' (comma).....N+4	=>	1/12 inch linefeed
or		
'0' (zero).....N+4	=>	Printer speed of 180 cps
'1' .....	=>	Printer speed of 150 cps
'2' .....	=>	Single pass printing
'3' .....	=>	Letter quality printing
if column N+5 is used.		
'1' .....	=>	Horizontal pitch of 10 cpi
'2' .....	=>	Horizontal pitch of 12 cpi
'3' .....	=>	Horizontal pitch of 13.3 cpi
'4' .....	=>	Horizontal pitch of 17.1 cpi (Draft only)
'5' .....	=>	Horizontal pitch of 15 cpi (Draft only)

## LIST OF SPECIAL CHARACTERS

### COLUMN N+1 = TYPE OF CHARACTER OPTION

Two special characters are recognized by the program if they are found in column N+1 of the file. They are the '~' (tilde) and the '.' (period). The option is applied to the first N columns of the current line. The printer is reset to STANDARD characters after the line is printed.

#### a) The '~' (tilde) calls ITALIC characters option

Next lines show italic characters available compared to standard :

```

abcdefghijklmnopqrstuvwxyz1234567890-='[];'\'./
abcdefghijklmnopqrstuvwxyz1234567890-='[];'\'./

ABCDEFGHIJKLMNOPQRSTUVWXYZ!@#%&^&*()_+~{}:"|<>?
ABCDEFGHIJKLMNOPQRSTUVWXYZ!@#%&^&*()_+~{}:"|<>?

```

#### b) The '.' (period) calls SCIENTIFIC characters option

Next lines show scientific characters available compared to standard :

```

abcdefghijklmnopqrstuvwxyz1234567890-='[];'\'./
αΒΨΦε>ληιϋκωμν↓ργθστξ†δχυζ¹²³⁴⁵⁶⁷⁸⁹⁰_@'..x^;`''

ABCDEFGHIJKLMNOPQRSTUVWXYZ!@#%&^&*()_+~{}:"|<>?
∇|Υφ÷<∧Π}∫§Ω∩~†ℓΓΘΣ™ΞαΔωΤ≈√≡←→±:Σ⊙()_+ ~ °•{Ππ(

```

### COLUMN N+2 = CENTRE LINE OPTION

The '\*' (asterisk) is recognized by the program if it is located in column N+2 of the file.

The character string on the current line will be centered assuming that the string starts in column one and ends with the last non-blank character of the same line.

To centre a formula that covers more than one line, the '`' (grave accent) is used as a right delimiter and is not printed.

For example :

	N+ 1	2	3	4	5	<= Column number
P = -	[ ]	[*]	[ ]	[ ]	[ ]	
A + 1.414 A`	[ ]	[ ]	[ ]	[ ]	[ ]	
12&	[ ]	[*]	[ ]	[ ]	[ ]	
i )	[ ]	[*]	[ ]	[ ]	[ ]	
i=0	[ ]	[*]	[ ]	[ ]	[ ]	

becomes :

$$P_1 = -\omega^2 \sum_{j=0}^i A_j + 1.414 A_i$$

COLUMN N+3 = UNDERLINE OPTION

The '\_' (underscore) is recognized if it is located in the column N+3 of the file. The underscore may be used as delimiter in columns 1 to N.

A character string of the current line will be underlined with the following conditions :

- 1) There is no '\_' from column 1 to N of the current line. In that case the string will be underlined from the first non-blank character met until the last non-blank of the current line.

For example :

	N+ 1	2	3	4	5	<= Column number
This has to be underlined.	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
become :						
<u>This has to be underlined.</u>						

- 2) Only one '\_' is found from column 1 to N of the current line. If there are non-blank characters to the right of delimiter, there should not be any non-blank character to its left. In that case, the underlining will start right after the '\_' until column N of the current line. The delimiter will not be printed.

For example :

Underline this :	N+ 1	2	3	4	5	<= Column number
_ until column N	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
become :						
<u>Underline this : until column N</u>						

If there are non-blank characters to the left of delimiter, there should not be any non-blank character to its right. In that case, the underlining will start on column 1 until the position preceding the '\_' of the current line. The delimiter will not be printed.

For example :

Underline this until the delimiter mark.	N+ 1	2	3	4	5
_	[ ]	[ ]	[ ]	[ ]	[ ]
become :					
<u>Underline this until the delimiter mark.</u>					

- 3) Two '\_' are found from column 1 to N of the current line. In this case, a line will be simply drawn between the two delimiters. Non-blank characters should not be found outside the limits. Neither delimiters will be printed.

For example :

Draw the following line :	N+ 1	2	3	4
_	[ ]	[ ]	[ ]	[ ]
become :				
Draw the following line : _____				

COLUMN N+4 = LINEFEED CONTROL OPTION

#### COLUMN N+4 = LINEFEED CONTROL OPTION

Two special characters are recognized by the program if they are found in column N+4 of the file. They are the '+' (plus sign) and the ',' (comma). This column may be used for another purpose if column N+5 is used.

##### a) The '+' (plus sign) allows overprinting

If the '+' (plus sign) is in column N+4, the current line will be printed without any preceding linefeed. This option is used generally when a multi-mode line has to be included in the document.

For example :

		N+ 1	2	3	4
MANY	can be inserted	[	]	]	]
modes		[~]	]	]	[+]
	simultaneously	[~]	]	]	[_][+]
	SAME	[	]	]	[_][+]
	Pihe	[.]	]	]	[+]

become :

MANY *modes* can be inserted simultaneously on the SAME line

##### b) The ',' (comma) calls the 1/12 inch linefeed

If the ',' (comma) is found in column N+4, the current line will be printed after a linefeed of exactly 1/12 inch. This option is used generally to introduce superscripts or subscripts. See example page 2. The line counter is incremented by 0.3334 each time this option is applied. However, that increment will not generate any form feed.

#### COLUMN N+5 = SET HORIZONTAL PITCH AND PRINTER SPEED

Five special characters are recognized by the program if they are found in column N+5 of the file. These characters are '1', '2', '3', '4' and '5'. Each of them represent a different option. The options represented by column N+4 will not be selected if column N+5 is not used :

N+4

- [0] => Select printer speed to 180 cps
- [1] => Select printer speed to 150 cps
- [2] => Select printer mode to single-pass printing
- [3] => Select printer mode to letter quality printing
- [ ] => No effect on the current printing speed or mode

These options are applied on columns 1 to N of the current line and are not reset for the following lines.

a) Option '1' sets horizontal pitch to 10 cpi

For exemple :

This line is printed in 10 characters per inch.  
become :

N+ 1 2 3 4 5  
[~][\*][\_][3][1]

This line is printed in 10 characters per inch.

b) Option '2' sets horizontal pitch to 12 cpi

For exemple :

This is the STANDARD 12 characters per inch.  
become :  
This is the STANDAND 12 characters per inch.

N+ 1 2 3 4 5  
[ ][ ][ ][ ][2]

c) Option '3' sets horizontal pitch to 13.3 cpi

For exemple :

This line is single-pass printed in 13.3 cpi.  
become :

N+ 1 2 3 4 5  
[ ][\*][ ][2][3]

This line is single-pass printed in 13.3 cpi.

d) Option '4' sets horizontal pitch to 17.1 cpi

For exemple :

The 17.1 cpi option can be used in draft printing only.  
become :

N+ 1 2 3 4 5  
[ ][ ][ ][1][4]

The 17.1 cpi option can be used in draft printing only.

e) Option '5' sets horizontal pitch to 15 cpi

For exemple :

The 15 cpi option can be used in draft printing only.  
become :

N+ 1 2 3 4 5  
[ ][ ][ ][0][5]

The 15 cpi option can be used in draft printing only.

Options '1', '2' and '3' can be used with the quality or the non-quality printing features while options '4' and '5' can be used only with non-quality printing.

Note on mixed mode printing using horizontal pitch option

Width of printed characters may vary along a single line using the overprinting option. The user should then gauge the space need to properly insert the new character string.

For exemple :

	N+ 1	2	3	4	5
different	[ ]	[ ]	[ ]	[ ]	[ ]
This inserted word is of	[ ]	[ ]	[ ]	[ ]	[ ]
size.	[ ]	[ ]	[ ]	[ ]	[ ]
becomes :					
This inserted word is of	[ ]	[ ]	[ ]	[ ]	[ ]
<u>different</u> size.	[ ]	[ ]	[ ]	[ ]	[ ]

The centre line option may not work if used with a non-standard horizontal pitch.

Any other option can be used with the horizontal pitch option without any problem.

ACKNOWLEDGMENT

This software constitutes an expansion of the original program NEWSSET developed by J. Halpenny



```

C F77 SCR,SCR=SCR/-SF
C TKB SCR=SCR,DLO:[1,1]F4POTS,OLB/LB
C
C ***** SCR.FTN *****
C This program is used to print Scientific Reports on the Qantex
C Model 7030 Printer. Printing mode selection responses are for
C Main FROM 884804, Rev F and higher Rev's.
C The program recognize 12 special characters and the position they
C hold within columns ML+1 to ML+5. ML is specified by the user in
C the input file. The maximum length of a text line is 123 characters.
C
0001 PROGRAM SCR
0002 BYTE BUFF(132),INFILE(20),OUTFIL(20),TEMP(129),SPEEDI
0003 BYTE OP1,OP2,OPP(6),TOP(6)
C BUFF may be extended to 132 characters with the underline option
C and the control character.
0004 INTEGER TAB(16),PAGE,NBOT(6)
0005 REAL LPP(6),BOT(6)
0006 EXTERNAL LEN
0007 DATA TAB/1,9,17,25,33,41,49,57,65,73,81,89,105,113,121,129/
0008 DATA OPP/'4','5','6','1','2','3'/
0009 DATA LPP/'22','33','44','66','88','132'// !=11*(LINE/INCH)
0010 DATA TOP/'2','3','4','4','4','4'/
0011 DATA BOT/'19','28','38','58','80','119'// !=TOP+NBOT
0012 DATA NBOT/17, 25, 34, 54, 76, 115/
0013 TYPE 30
0014 30 FORMAT (' Scientific Set Print. Filename? ', $)
0015 ACCEPT 1, INFILE
0016 1 FORMAT (20A1)
0017 DO 16 I=20,4,-1 !Move filename 3 cases right
0018 INFILE(I)=INFILE(I-3)
0019 16 CONTINUE
0020 INFILE(1)='S' !Inserting SY: before the filename
0021 INFILE(2)='Y'
0022 INFILE(3)=':'
0023 INFILE(20)=0 !End on null
0024 OPEN (UNIT=2,NAME=INFILE,TYPE='OLD',READONLY)
0025 READ (2,*) MLO
0026 IF (MLO.GT.123) THEN
0027 TYPE *, ' Maximum length of a text line is 123 characters.'
0028 TYPE *, ' Reduce the value in the input file.'
0029 CALL EXIT
0030 END IF
0031 TYPE 133
0032 133 FORMAT (' Do you want page number (Y/N) ? ', $)
0033 ACCEPT 4, OP1
0034 4 FORMAT (A1)
0035 IF (OP1.EQ.'Y'.OR.OP1.EQ.'y') OP1 = 'Y'
0036 124 TYPE 135
0037 135 FORMAT ('0',9X,'LINES LINES',/, ' OPTION /INCH /PAGE',/,
1 ' 1',8X,'2',7X,'17',/, ' 2',8X,'3',7X,'25',/, ' 3',8X,'4',
2 7X,'34',/, ' 4',8X,'6',7X,'54 (=Default',/, ' 5',8X,'8',7X,
3 '73',/, ' 6',7X,'12 115',//.
4 ' Insert lines/page OPTION : ', $)
0038 ACCEPT 5, OP2

```

```

0039 5      FORMAT (I1)
0040      IF (OP2,EQ,0) OP2 = 4
0041      IF (OP2,GT,6) THEN
0042          TYPE 136
0043 136     FORMAT (' **** OUT OF RANGE OPTION !')
0044          GOTO 124
0045      END IF
0046      ML1 = ML0 + 1      !\
0047      ML2 = ML0 + 2      !: These five integers
0048      ML3 = ML0 + 3      !: represents columns containing
0049      ML4 = ML0 + 4      !: special characters.
0050      ML5 = ML0 + 5      !/
0051      L1=0
0052      PAGE = 0          !Page numbering
0053      N=0
0054 20      N=N+1
0055          OUTFIL(N)=INFILE(N)      !Keep same name as INFILE
0056          IF (INFILE(N),NE,'.' AND,INFILE(N),NE,' ') GOTO 20
0057          OUTFIL(N)='.'
0058          OUTFIL(N+1)='Q'          !Add the suffix .QTX
0059          OUTFIL(N+2)='T'
0060          N = N + 3
0061          OUTFIL(N)='X'
0062          OPEN (UNIT=1,NAME=OUTFIL,TYPE='NEW')
0063          TYPE 37, (OUTFIL(I), I=4,N)
0064 37      FORMAT (' Creating output file ',20A1)
C *****
0065      XLINE = 0.          !Number of lines per page
0066      SPEED = '3'
C Default setting will be standard font with 12 cpi in quality letter.
0067      NFORM = 2
0068      IF (OP2,EQ,6) NFORM = 3
0069      WRITE (1,132) 27,SPEED,27,27,27,VPP(OP2),27,LPP(OP2),
* 27,TOP(OP2),BOT(OP2)
C In order ESC[n!p : Speed in character per second (default n=3)
C ESC[2w : Horizontal pitch (default)
C ESC[l!m : Cubic print font (default)
C ESC[nz : Vertical pitch (default n=1)
C ESC[nt : Form length (default n=66)
C ESC[n!mr : Top and bottom margin (default n=4, m=58)
0070 132     FORMAT ('+',A1,'[',A1,'!p',A1,'[2w',A1,'[l!m',A1,'[',A1,'z',
* A1,'[',A(NFORM),'t',A1,'[',A1,';',A(NFORM),'r')
0071      GOTO 121
0072 24      XLINE = XLINE + 1.
0073          WRITE (1,31) (BUFF(I),I=1,LENG)
0074 31      FORMAT (' ',131A1)
0075          DO 110 I=ML1,ML5      !Erase special columns
0076          BUFF(I) = ' '
0077 110     CONTINUE
0078          IF (XLINE,GE,NBOT(OP2)) THEN      !End of a page
0079              XLINE = 0.
0080              IF (OP1,EQ,'Y') THEN
0081                  PAGE = PAGE + 1
0082                  WRITE (1,130) PAGE
C Write the FORTRAN control char. '1' in column 1 and number the page

```

```

0083 130      FORMAT ('1', <ML0+2>X, I3)
0084      ELSE
0085      WRITE (1,134)
0086 134      FORMAT ('1')
0087      END IF
0088      END IF
0089 21      IF (L1.EQ.0) GOTO 121
0090      IF (SPEED.EQ.'2'.OR.SPEED.EQ.'3') THEN
0091      WRITE (1,35) 27,SPEED,27
0092 35      FORMAT ('+',A1,'[',A1,'!p',A1,'[11m') !Set local characters
0093      ELSE
0094      WRITE (1,231) 27,27,SPEED
0095 231      FORMAT ('+',A1,'[11m',A1,'[',A1,'!p')
0096      END IF
0097      L1 = 0
0098 121     READ (2,2,END=23) (TEMP(I),I=1,ML5)
0099 2       FORMAT (128A1)
0100      IF (TEMP(1).EQ.12) THEN          !Formfeed in text
0101      XLINE = 0.
0102      IF (OP1.EQ.'Y') THEN
0103      PAGE = PAGE + 1
0104      WRITE (1,130) PAGE
C       Write the FORTRAN control char. '1' in column 1 and number the page
0105      ELSE
0106      WRITE (1,134)
0107      END IF
0108      GOTO 121
0109      END IF
0110      LENGT = LEN(TEMP,ML5)
C       *****
C       Repracement of TABs by blanks
C       *****
0111      N = 0
0112      DO 17 I=1,LENGT          !Position on TEMP
0113      N = N + 1                !Position on BUFF
0114      IF (TEMP(I).EQ.9) THEN ! 9 is the ASCII code for a horiz. tab
0115      DO 18 J=1,15
0116      IF (N.GE.TAB(J).AND.N.LT.TAB(J+1)) GOTO 25
0117 18      CONTINUE
0118 25      LIM = TAB(J+1) - 1
0119      DO 19 J=N,LIM
0120      BUFF(J) = ' '
0121 19      CONTINUE
0122      N = LIM
0123      ELSE
0124      BUFF(N) = TEMP(I)
0125      END IF
0126 17      CONTINUE
0127      LENGT = N
0128      IF (LENGT.LT.ML1) THEN !No presence of special character
0129      LENG = LENGT
0130      GOTO 24
0131      END IF
C       *****
C       COLUMN ML+5 = HORIZONTAL PITCH

```

```

C *****
0132 IF (LENGT.EQ.ML5) THEN !Fixe horizontal pitch and printer speed
0133 IF (BUFF(ML4).GT.47) SPEED = BUFF(ML4) !ASCII code dec. value
0134 IF (SPEED.EQ.'2'.OR.SPEED.EQ.'3') THEN
0135 WRITE (1,39) 27,BUFF(ML5),27,SPEED,27
0136 39 FORMAT ('+',A1,'[',A1,'w',A1,'[',A1,'!p',A1,'[11m')
0137 ELSE IF (SPEED.EQ.'0'.OR.SPEED.EQ.'1') THEN
0138 WRITE (1,230) 27,BUFF(ML5),27,27,SPEED
0139 230 FORMAT ('+',A1,'[',A1,'w',A1,'[11m',A1,'[',A1,'!p')
0140 END IF
0141 END IF
C *****
0142 LENG = LEN(BUFF,ML0) !Actual length of text line
0143 L4 = 0
0144 M2 = 1 !Presumed start of text line
C *****
C COLUMN ML+1 = TYPE OF CHARACTER OPTION
C *****
0145 IF (BUFF(ML1).NE.' ') THEN !Limit to one test on most line
0146 IF (BUFF(ML1).EQ.'~') THEN !Set Italic Characters
0147 IF (SPEED.EQ.'2'.OR.SPEED.EQ.'3') THEN
0148 WRITE(1,36) 27,SPEED,27
0149 36 FORMAT ('+',A1,'[',A1,'!p',A1,'[13m')
0150 ELSE
0151 WRITE(1,137) 27,27,SPEED
0152 137 FORMAT ('+',A1,'[13m',A1,'[',A1,'!p')
0153 END IF
0154 L1=1 !Flag for reset
0155 ELSE IF (BUFF(ML1).EQ.'.') THEN !Set Scientific Characters
0156 IF (SPEED.EQ.'2'.OR.SPEED.EQ.'3') THEN
0157 WRITE (1,34) 27,SPEED,27
0158 34 FORMAT ('+',A1,'[',A1,'!p',A1,'[14m')
0159 ELSE
0160 WRITE(1,138) 27,27,SPEED
0161 138 FORMAT ('+',A1,'[14m',A1,'[',A1,'!p')
0162 END IF
0163 L1=1 !Flag for reset
0164 END IF
0165 IF (LENG.EQ.ML1) GOTO 24
0166 END IF
C *****
C COLUMN ML+2 = Centre line feature (always centre before underline)
C *****
0167 IF (BUFF(ML2).EQ.'*') THEN !Centre option
0168 M1=LENG
0169 IF (BUFF(M1).EQ.'') BUFF(M1)=' ' !Take off delimiter sign
0170 M2 = (ML0 - M1)/2
0171 IF (M2.LT.1) THEN !No move. Go look NOW for next option
0172 IF (LENGT.EQ.ML2) GOTO 24
0173 M2 = 1
0174 GOTO 29
0175 END IF
0176 LENG = M1 + M2
0177 DO 12 I=M1,1,-1
0178 BUFF(I+M2)=BUFF(I) !Move all string M2 spaces right.

```

```

0179 12      CONTINUE
0180          DO 13 I=1,M2
0181          BUFF(I)=' '           !Erase leading characters
0182 13      CONTINUE
0183          END IF
C *****
0184 29      IF (BUFF(ML4).NE.' ') THEN           !Preliminary glance into column ML+4
0185          IF (BUFF(ML4).EQ.'+') THEN         !Save BUFF(ML4) before moving
0186              L4 = 1                         ! string to the right
0187          ELSE IF (BUFF(ML4).EQ.',') THEN
0188              L4 = 2
0189          END IF
0190          END IF
C *****
C          COLUMN ML+3 = UNDERLINE OPTION
C *****
0191          IF (BUFF(ML3).EQ.'_') THEN         !Underline option
0192          DO 14 I=LENG,M2,-1                 !Move all string 4 spaces right.
0193          BUFF(I+4)=BUFF(I)
0194 14      CONTINUE
0195          LENG = LENG + 4                   !New actual length of text line
0196          M2 = M2 + 4                       !new presumed start of text line
0197          N=0
0198          N1=0
0199          N2=0
0200          DO 10 I=M2,LENG
0201          IF (BUFF(I).NE.'_') GOTO 10
0202          N=N+1                             !Number of '_' on the line
0203          IF (N1.NE.0) GOTO 28
0204          N1=I                               !Position of the first '_' on the line
0205          GOTO 10
0206 28      N2=I                               !Position of the second '_' on the line
0207          GOTO 120
0208 10      CONTINUE
0209 120     IF (N.EQ.0) THEN                   !No underscore on the line
0210          DO 11 I=M2,LENG
0211          N1=I                               !Position of the first non-blank character
0212          IF (BUFF(I).NE.' ') GOTO 26
0213 11      CONTINUE
0214          LENG = 1
0215          GOTO 27                             !To write BUFF without underlining
0216 26      N2=LENG
0217          N1=N1-4
0218          BUFF(N1)=27                         !Open underline option
0219          BUFF(N1+1)='['
0220          BUFF(N1+2)='4'
0221          BUFF(N1+3)='m'
0222          N2=N2+1
0223          BUFF(N2)=27                         !Close underline option
0224          BUFF(N2+1)='['
0225          BUFF(N2+2)='0'
0226          BUFF(N2+3)='m'
0227          LENG = N2 + 3
0228          ELSE IF (N.EQ.1) THEN             !One underscore on the line
0229          DO 15 I=M2,LENG

```

```

0230             IF (BUFF(I).EQ.' ') GOTO 15
0231             GOTO 22
0232     15      CONTINUE
C             If just one underscore is on the line, non-blank characters should be
c             on one side only of the delimiter.
0233     22      IF (I.NE.N1) THEN
0234             M2 = M2 - 4
0235             BUFF(M2)=27
0236             BUFF(M2+1)='[' !Open underline option on left side of '_'
0237             BUFF(M2+2)='4'
0238             BUFF(M2+3)='m'
0239             BUFF(N1)=27 !Close underline option
0240             BUFF(N1+1)='['
0241             BUFF(N1+2)='0'
0242             BUFF(N1+3)='m'
0243             LENG = N1 + 3
0244     ELSE
0245             N1=N1-3
0246             BUFF(N1)=27 !Open underline option on right side of '_'
0247             BUFF(N1+1)='['
0248             BUFF(N1+2)='4'
0249             BUFF(N1+3)='m'
0250             BUFF(ML1) = ' '
0251             BUFF(ML2) = ' '
0252             BUFF(ML3) = ' '
0253             BUFF(ML4) = ' '
0254             BUFF(ML5)=27 !Close underline option
0255             BUFF(ML5+1)='['
0256             BUFF(ML5+2)='0'
0257             LENG = ML5 + 3
0258             BUFF(LENG)='m'
0259     END IF
0260     ELSE IF (N.EQ.2) THEN !Two underscores on the line
0261     N1=N1-3
C             When there are two underscore on the same line, there should not be
C             non-blank characters outside the limits defined by underscores.
0262             BUFF (N1)=27 !Open underline option
0263             BUFF (N1+1)='['
0264             BUFF (N1+2)='4'
0265             BUFF (N1+3)='m'
0266             BUFF (N2)=27 !Close underline option
0267             BUFF (N2+1)='['
0268             BUFF (N2+2)='0'
0269             BUFF (N2+3)='m'
0270             LENG = N2 + 3
0271     END IF
0272     END IF
C             *****
C             COLUMN ML+4 = LINEFEED CONTROL OPTION
C             *****
0273     27      IF (L4.EQ.0) GOTO 24 !No linefeed control
0274             IF (L4.EQ.1) THEN !Overprint line
0275             WRITE (1,32) (BUFF(I), I=1,LENG)
0276     32      FORMAT ('+',131A1)
0277             GOTO 21

```

```
0278         ELSE                                !Linefeed 1/12*
0279             WRITE (1,33) 27
0280     33     FORMAT ('+',A1,'[4!s')
0281             WRITE (1,32) (BUFF(I),I=1,LENG)
0282             XLINE = XLINE + 0.3334
0283             GOTO 21
0284         END IF
0285     23     TYPE 38
0286     38     FORMAT (' Do you want to print now on Qantex (Y/N) ? ', $)
0287         ACCEPT 4, OP1
0288         IF (OP1.EQ.'Y'.OR.OP1.EQ.'y') THEN
0289             OPEN (UNIT=6,NAME='LP1:',TYPE='NEW')
0290             WRITE(6,131) 27,27,27                !Standard options
0291     131     FORMAT ('+',A1,'[12;132s',A1,'(B',A1,'[0!y')
0292             REWIND 1
0293             MLO = MLO + 9
0294     123     READ (1,3,END=122) (BUFF(I),I=1,MLO)
0295     3       FORMAT (132A1)
0296             WRITE(6,3) (BUFF(I),I=1,MLO)
0297             GOTO 123
0298     122     WRITE(6,139) 12,12,27                !Send two formfeeds and reset Qantex
0299     139     FORMAT ('+',3A1,'c')
0300         ELSE
0301             CALL EXIT
0302         END IF
0303     END
```

PROGRAM SECTIONS

Number	Name	Size	Attributes
1	\$CODE1	006666 1755	RW,I,CON,LCL
2	\$PDATA	001604 450	RW,D,CON,LCL
3	\$IDATA	000010 4	RW,D,CON,LCL
4	\$VARS	000704 226	RW,D,CON,LCL
5	\$TEMPS	000004 2	RW,D,CON,LCL

VARIABLES

Name	Type	Address	Name	Type	Address	Name	Type	Address	Name	Type	Address	Name	Type	Address
I	I*2	4-000632	J	I*2	4-000666	LENG	I*2	4-000662	LENGT	I*2	4-000664	LIM	I*2	4-000670
L1	I*2	4-000650	L4	I*2	4-000672	ML0	I*2	4-000634	ML1	I*2	4-000636	ML2	I*2	4-000640
ML3	I*2	4-000642	ML4	I*2	4-000644	ML5	I*2	4-000646	M1	I*2	4-000676	M2	I*2	4-000674
N	I*2	4-000652	NFORM	I*2	4-000660	N1	I*2	4-000700	N2	I*2	4-000702	OP1	L*1	4-000456
OP2	L*1	4-000457	PAGE	I*2	4-000534	SPEED	L*1	4-000455	XLIN	R*4	4-000654			

ARRAYS

Name	Type	Address	Size	Dimensions
BOT	R*4	4-000602	000030	12 (6)
BUFF	L*1	4-000000	000204	66 (132)
INFILE	L*1	4-000204	000024	10 (20)
LPP	R*4	4-000552	000030	12 (6)
NBOT	I*2	4-000536	000014	6 (6)
OUTFIL	L*1	4-000230	000024	10 (20)
TAB	I*2	4-000474	000040	16 (16)
TEMP	L*1	4-000254	000201	65 (129)
TOP	L*1	4-000466	000006	3 (6)
VPP	L*1	4-000460	000006	3 (6)

LABELS

Label	Address	Label	Address	Label	Address	Label	Address	Label	Address
1'	2-000046	2'	2-000724	3'	2-001314	4'	2-000120	5'	2-000430
10	1-004536	11	**	12	**	13	**	14	**
15	1-005116	16	**	17	**	18	**	19	**
20	1-000674	21	1-001664	22	1-005144	23	1-006142	24	1-001350
25	1-002464	26	1-004676	27	1-005634	28	1-004522	29	1-004206
30'	2-000000	31'	2-000624	32'	2-001154	33'	2-001162	34'	2-001100
35'	2-000650	36'	2-001024	37'	2-000472	38'	2-001176	39'	2-000730
110	**	120	1-004560	121	1-002062	122	1-006536	123	1-006336
124	1-000426	130'	2-000632	131'	2-001256	132'	2-000526	133'	2-000052
134'	2-000644	135'	2-000124	136'	2-000434	137'	2-001052	138'	2-001126
139'	2-001320	230'	2-000766	231'	2-000676				



FUNCTIONS AND SUBROUTINES REFERENCED

EXIT LEN OPEN\*

Total Space Allocated = 011412 2437

SCR.FTN:112

/F77/OP/TR:BLOCKS/WR

```
0001      FUNCTION LEN(BUFF,L)  !# of bytes without trailing blanks
0002      BYTE BUFF(132)        !Can be extended to 132 characters with
                                the underline option and control char.
                                C
0003      DO 10 LEN=L,1,-1
0004      IF (BUFF(LEN).NE.' ') RETURN  !First non blank character.
0005  10  CONTINUE
0006      LEN=1                  !Line must have one character.
0007      END
```

PROGRAM SECTIONS

Number	Name	Size	Attributes
1	\$CODE1	000146	51 RW,I,CON,LCL
3	\$IDATA	000014	6 RW,D,CON,LCL

ENTRY POINTS

Name	Type	Address	Name	Type	Address	Name	Type	Address	Name	Type	Address	Name	Type	Address
LEN	I*2	3-000000												

VARIABLES

Name	Type	Address	Name	Type	Address	Name	Type	Address	Name	Type	Address	Name	Type	Address
L	I*2	F-000004*												

ARRAYS

Name	Type	Address	Size	Dimensions
BUFF	L*1	F-000002*	000204	66 (132)

LABELS

Label	Address	Label	Address	Label	Address	Label	Address	Label	Address
10	**								

Total Space Allocated = 000162 57

No FPP Instructions Generated