

FLGS
V04-

.....

FFFFFFFF		JJ	NN	NN	FFFFFFFF	NN	NN		JJ
FFFFFFFF		JJ	NN	NN	FFFFFFFF	NN	NN		JJ
FF		JJ	NN	NN	FF	NN	NN		JJ
FF		JJ	NN	NN	FF	NN	NN		JJ
FF		JJ	NNNN	NN	FF	NNNN	NN		JJ
FF		JJ	NNNN	NN	FF	NNNN	NN		JJ
FFFFFFFF		JJ	NN	NN	FFFFFFFF	NN	NN		JJ
FFFFFFFF		JJ	NN	NN	FFFFFFFF	NN	NN		JJ
FF	JJ	JJ	NN	NNNN	FF	NN	NNNN	JJ	JJ
FF	JJ	JJ	NN	NNNN	FF	NN	NNNN	JJ	JJ
FF	JJ	JJ	NN	NN	FF	NN	NN	JJ	JJ
FF	JJ	JJ	NN	NN	FF	NN	NN	JJ	JJ
FF		JJ	NN	NN	FF	NN	NN		JJ
FF	JJJJJJ		NN	NN	FF	NN	NN	JJJJJJ	
FF	JJJJJJ		NN	NN	FF	NN	NN	JJJJJJ	

```

LL               IIIIII               SSSSSSSS
LL               IIIIII               SSSSSSSS
LL               II                    SS
LL               II                    SS
LL               II                    SS
LL               II                    SS
LL               II                    SSSSSS
LL               II                    SSSSSS
LL               II                    SS
LL               II                    SS
LL               II                    SS
LL               II                    SS
LLLLLLLLLLLLLL  IIIIII               SSSSSSSS
LLLLLLLLLLLLLL  IIIIII               SSSSSSSS

```



```
1 0001 0 MODULE fjfnfj ( IDENT = 'V04-000'
2 P 0002 0 %BLISS32[, ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE,
3 0003 0 NONEXTERNAL = LONG_RELATIVE)]
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1 *****
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
11 0011 1 * ALL RIGHTS RESERVED.
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
18 0018 1 * TRANSFERRED.
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
22 0022 1 * CORPORATION.
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
26 0026 1 *
27 0027 1 *
28 0028 1 *****
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 Processes .AUTOFILL, .NO AUTOFILL, .FILL, .JUSTIFY, .NO FILL,
36 0036 1 .NO JUSTIFY, .KEEP, .NO KEEP, .AUTOJUSTIFY, and .NO AUTOJUSTIFY
37 0037 1
38 0038 1 ENVIRONMENT: Transportable
39 0039 1
40 0040 1 AUTHOR: R.W.Friday CREATION DATE: May, 1978
41 0041 1
42 0042 1
```

44
45
46
47
48
49
50
51
52
53

```
0043 1 %SBTTL 'Revision History'
0044 1 |
0045 1 |   MODIFIED BY:
0046 1 |
0047 1 |           005          RER00005
0048 1 |                   Global ed
0049 1 |                   copyright
0050 1 |
0051 1 |   --
0052 1 |
```

005 RER00005 Ron Randall 07-Mar-1983
Global edit of all modules. Updated module names, idents,
copyright dates. Changed require files to BLISS library.

Module Level Declarations

```
: 55      0053 1 %SBTTL 'Module Level Declarations'
: 56      0054 1
: 57      0055 1
: 58      0056 1 : TABLE OF CONTENTS:
: 59      0057 1
: 60      0058 1
: 61      0059 1 : INCLUDE FILES:
: 62      0060 1
: 63      0061 1 LIBRARY 'NXPOR:XPOR';      ! XPORT Library
: 64      0062 1 REQUIRE 'REQ:RNODEF';      ! RUNOFF variant definitions
: 65      0193 1
: 66      U 0194 1 %IF DSRPLUS %THEN
: 67      U 0195 1 LIBRARY 'REQ:DPLLIB';      ! DSRPLUS BLISS Library
: 68      0196 1 %ELSE
: 69      0197 1 LIBRARY 'REQ:DSRLIB';      ! DSR BLISS Library
: 70      0198 1 %FI
: 71      0199 1
: 72      0200 1
: 73      0201 1 : EXTERNAL REFERENCES:
: 74      0202 1
: 75      0203 1 EXTERNAL
: 76      0204 1     GCA : GCA_DEFINITION,
: 77      0205 1     SCA : SCA_DEFINITION;
: 78      0206 1
```

Module Level Declarations

```

80 0207 1 GLOBAL ROUTINE FJNFNJ (HANDLER_CODE) : NOVALUE =
81 0208 1
82 0209 1 ++
83 0210 1 FUNCTIONAL DESCRIPTION:
84 0211 1
85 0212 1 FJNFNJ makes sure that the fill and justification
86 0213 1 modes get set correctly.
87 0214 1 It also controls whether or not empty records get kept when
88 0215 1 in .NO FILL mode.
89 0216 1
90 0217 1 FORMAL PARAMETERS:
91 0218 1
92 0219 1 HANDLER_CODE indicates which command is to be processed
93 0220 1
94 0221 1 IMPLICIT INPUTS: None
95 0222 1
96 0223 1 IMPLICIT OUTPUTS: None
97 0224 1
98 0225 1 ROUTINE VALUE:
99 0226 1 COMPLETION CODES: None
100 0227 1
101 0228 1 SIDE EFFECTS: None
102 0229 1 --
103 0230 1
104 0231 2 BEGIN
105 0232 2 !SCA_FILL and SCA_JUSTIFY are TRUE for the respective functions.
106 0233 2 !SCA_CROCK is for compatibility with TOPS-10 RUNOFF. Basically,
107 0234 2 if you say .NO FILL, you "probably" don't want justification
108 0235 2 any more; on the other hand, when you say .FILL you probably
109 0236 2 do want justification, assuming you did not say .NO JUSTIFY.
110 0237 2 In other words, SCA_CROCK is used to remember whether or not you
111 0238 2 want justification when you say .FILL.
112 0239 2
113 U 0240 2 XIF DSRPLUS XTHEN
114 UU 0241 2 !.NO AUTOJUSTIFY will now prevent .HL, CHAPTER, and APPENDIX from
115 U 0242 2 !resetting fill and justify after their operation.
116 0243 2 XFI
117 0244 2
118 0245 2 SELECTONE .HANDLER_CODE OF
119 0246 2 SET
120 0247 2
121 0248 2 [H_FILL] :
122 0249 2 BEGIN
123 0250 2 SCA_FILL = TRUE;
124 0251 2 SCA_JUSTIFY = .SCA_CROCK;
125 0252 2 SCA_KER = FALSE;
126 0253 2 END;
127 0254 2
128 0255 2 [H_JUSTIFY] :
129 0256 2 BEGIN
130 0257 2 SCA_JUSTIFY = TRUE;
131 0258 2 SCA_CROCK = TRUE;
132 0259 2 END;
133 0260 2
134 0261 2 [H_NO_FILL] :
135 0262 2 BEGIN
136 0263 2 SCA_JUSTIFY = FALSE;

```

!Empty records have no significance.


```
137 G264 3          SCA_FILL = FALSE;
138 0265 3          SCA_KER = .GCA_KEEP;          !Empty records generate blank lines if user said .KEEP.
139 0266 3          END;
140 0267 3
141 0268 3      [H_NO_JUSTIFY] :
142 0269 3          BEGIN
143 0270 3          SCA_JUSTIFY = FALSE;
144 0271 3          SCA_CROCK = FALSE;
145 0272 3          END;
146 0273 3
147 0274 3      [H_KEEP] :
148 0275 3          BEGIN
149 0276 3          GCA_KEEP = TRUE;          !User want's empty records kept in .NO FILL mode
150 0277 3
151 0278 3          !Turn on .KEEP mode if already in .NO FILL mode, so the order of the
152 0279 3          !commands is not important.
153 0280 3          IF NOT .SCA_FILL
154 0281 3          THEN
155 0282 3              SCA_KER = TRUE;
156 0283 3          END;
157 0284 3
158 0285 3      [H_NO_KEEP] :
159 0286 3          BEGIN
160 0287 3          GCA_KEEP = FALSE;          !Discard empty records in .NO FILL mode
161 0288 3
162 0289 3          !Turn off .KEEP mode if already in .NO FILL mode, so the order of the
163 0290 3          !commands is not important.
164 0291 3          IF NOT .SCA_FILL
165 0292 3          THEN
166 0293 3              SCA_KER = FALSE;
167 0294 3          END;
168 0295 3
169 0296 3      [H_AUTOJUSTIFY] :
170 0297 3          BEGIN
171 0298 3          GCA_AUTOJUST = TRUE;
172 0299 3          END;
173 0300 3
174 0301 3      [H_NO_AUTOJUSTIF] :
175 0302 3          BEGIN
176 0303 3          GCA_AUTOJUST = FALSE;
177 0304 3          END;
178 0305 3      TES;
179 0306 3
180 0307 1      END;          !End of FJNFNJ
```

```
.TITLE FJNFNJ
.IDENT \V04-000\
```

```
.EXTRN GCA, SCA
```

```
.PSECT $CODE$,NOWRT,2
```

```
.ENTRY FJNFNJ, Save R2,R3
MOVAB GCA+204, R3
MOVAB SCA+104, R2
MOVL HANDLER_CODE, R0
```

```
000C 00000
53 00000000G EF 9E 00002
52 00000000G EF 9E 00009
50 04 AC D0 00010
```

```
: 0207
:
: 0245
```

00000043	8F		50	D1	00014	CMPL	R0, #67	:	0248	
			0B	12	0001B	BNEQ	1\$	:		
00	B2		01	D0	0001D	MOVL	#1, @SCA+104	:	0250	
FC	B2	08	B2	D0	00021	MOVL	@SCA+112, @SCA+100	:	0251	
			5D	11	00026	BRB	6\$	:	0252	
00000069	8F		50	D1	00028	1\$:	CMPL	R0, #105	:	0255
			09	12	0002F	BNEQ	2\$	:		
FC	B2		01	D0	00031	MOVL	#1, @SCA+100	:	0257	
08	B2		01	D0	00035	MOVL	#1, @SCA+112	:	0258	
			04	00039	RET			:	0245	
0000007B	8F		50	D1	0003A	2\$:	CMPL	R0, #123	:	0261
			0C	12	00041	BNEQ	3\$	:		
		FC	B2	D4	00043	CLRL	@SCA+100	:	0263	
		00	B2	D4	00046	CLRL	@SCA+104	:	0264	
1C	B2	00	B3	D0	00049	MOVL	@GCA+204, @SCA+132	:	0265	
			04	0004E	RET			:	0245	
00000094	8F		50	D1	0004F	3\$:	CMPL	R0, #148	:	0268
			07	12	00056	BNEQ	4\$	:		
		FC	B2	D4	00058	CLRL	@SCA+100	:	0270	
		08	B2	D4	0005B	CLRL	@SCA+112	:	0271	
			04	0005E	RET			:	0245	
0000006A	8F		50	D1	0005F	4\$:	CMPL	R0, #106	:	0274
			0D	12	00066	BNEQ	5\$	:		
00	B3		01	D0	00068	MOVL	#1, @GCA+204	:	0276	
	31	00	B2	E8	0006C	BLBS	@SCA+104, 9\$	:	0280	
1C	B2		01	D0	00070	MOVL	#1, @SCA+132	:	0282	
			04	00074	RET			:	0245	
00000093	8F		50	D1	00075	5\$:	CMPL	R0, #147	:	0285
			0B	12	0007C	BNEQ	7\$	:		
		00	B3	D4	0007E	CLRL	@GCA+204	:	0287	
		00	B2	E8	00081	BLBS	@SCA+104, 9\$	:	0291	
	1C	1C	B2	D4	00085	6\$:	CLRL	@SCA+132	:	0293
			04	00088	RET			:	0245	
	02		50	D1	00089	7\$:	CMPL	R0, #2	:	0296
			06	12	0008C	BNEQ	8\$	:		
FF44	D3		01	D0	0008E	MOVL	#1, @GCA+16	:	0298	
			04	00093	RET			:	0245	
00000073	8F		50	D1	00094	8\$:	CMPL	R0, #115	:	0301
			04	12	0009B	BNEQ	9\$	:		
		FF44	D3	D4	0009D	CLRL	@GCA+16	:	0303	
			04	000A1	9\$:	RET		:	0307	

; Routine Size: 162 bytes, Routine Base: \$CODE\$ + 0000

: 181 0308 1
: 182 0309 1 END
: 183 0310 0 ELUDOM

!End of module

PSECT SUMMARY

Name	Bytes	Attributes
------	-------	------------

;
; \$CODE\$

162 NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
;\$255\$DUA28:[SYSLIB]XPORT.L32;1	590	0	0	252	00:00.2
;\$255\$DUA28:[RUNOFF.SRC]DSRLIB.L32;1	1248	20	1	86	00:00.3

; COMMAND QUALIFIERS

; BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:FJNFNJ/OBJ=OBJ\$:FJNFNJ MSRC\$:FJNFNJ/UPDATE=(ENH\$:FJNFNJ)

; Size: 162 code + 0 data bytes
; Run Time: 00:04.2
; Elapsed Time: 00:18.1
; Lines/CPU Min: 4460
; Lexemes/CPU-Min: 15726
; Memory Used: 42 pages
; Compilation Complete

0341

AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY