

```
RRRRRRRRRRRRR      UUU      UUU      NNN      NNN      00000000      FFFFFFFFFFFFFFFF      FFFFFFFFFFFFFFFF  
RRRRRRRRRRRRR      UUU      UUU      NNN      NNN      00000000      FFFFFFFFFFFFFFFF      FFFFFFFFFFFFFFFF  
RRRRRRRRRRRRR      UUU      UUU      NNN      NNN      00000000      FFFFFFFFFFFFFFFF      FFFFFFFFFFFFFFFF  
RRR           RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR           RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR           RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR           RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR           RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR           RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR           RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRRRRRRRRRRRR      UUU      UUU      NNN      NNN      000           000      FFFFFFFFFFFFFFFF      FFFFFFFFFFFFFFFF  
RRRRRRRRRRRRR      UUU      UUU      NNN      NNN      000           000      FFFFFFFFFFFFFFFF      FFFFFFFFFFFFFFFF  
RRRRRRRRRRRRR      UUU      UUU      NNN      NNN      000           000      FFFFFFFFFFFFFFFF      FFFFFFFFFFFFFFFF  
RRR   RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR   RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR   RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR   RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR   RRR      UUU      UUU      NNN      NNN      000           000      FFF           FFF  
RRR           RRR      UUUUUUUUUUUUUUU      NNN      NNN      00000000      FFF           FFF  
RRR           RRR      UUUUUUUUUUUUUUU      NNN      NNN      00000000      FFF           FFF  
RRR           RRR      UUUUUUUUUUUUUUU      NNN      NNN      00000000      FFF           FFF
```

Syn

NDX

NDX

NUM

NUM

OPER

OUT

PAC

PAC

PAC

PAC

PAC

PAC

PAC

PAD

PAG

PAG

PAG

PAG

PAG

PAG

PER

PUT

RCO

RIN

RLI

RNO

RNO

RTY

SAV

STR

STR

STR

STR

STR

STR

STR

STR

STR

STR

STR

STR

STR

STR

STR

STR



```

FFFFFFFFF      IIIII
FFFFFFFFF      IIIII
FF            II
FF            II
FF            II
FF            II
FFFFFFF       II
FFFFFFF       II
FF            II
FF            II
FF            II
FF            II
FF            IIIII
FF            IIIII

GGGGGGGG      UU      UU      RRRRRRR      EEEEEEEEE
GGGGGGGG      UU      UU      RRRRRRR      EEEEEEEEE
GG            UU      UU      RR          RR      EE
GG            UU      UU      RR          RR      EE
GG            UU      UU      RR          RR      EE
GG            UU      UU      RRRRRRR      EEEEEEE
GG            UU      UU      RRRRRRR      EEEEEEE
GG      GGGGGG  UU      UU      RR      RR      EE
GG      GGGGGG  UU      UU      RR      RR      EE
GG            UU      UU      RR          RR      EE
GG            UU      UU      RR          RR      EE
GGGGGG      UUUUUUUUU  RR          RR      EEEEEEEEE
GGGGGG      UUUUUUUUU  RR          RR      EEEEEEEEE

```

```

LL            IIIII
LL            IIIII
LL            II
LL            II
LL            II
LL            II
LL            II
LL            II
LL            II
LL            II
LL            IIIII
LLLLLLLLLL   IIIII
LLLLLLLLLL   IIIII

SSSSSSSS
SSSSSSSS
SS
SS
SS
SS
SSSSSS
SSSSSS
SS
SS
SS
SS
SSSSSSSS
SSSSSSSS

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39

```

0001 0 MODULE figure ( IDENT = 'V04-000'
P 0002 0      %BLISS32 [ , ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE,
0003 0      ) =
0004 0      NONEXTERNAL = LONG_RELATIVE) ]
0005 1 BEGIN
0006 1
0007 1 *****
0008 1 *
0009 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0010 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0011 1 *  ALL RIGHTS RESERVED.
0012 1 *
0013 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0014 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0015 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0016 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0017 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SCFTWARE IS HEREBY
0018 1 *  TRANSFERRED.
0019 1 *
0020 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0021 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0022 1 *  CORPORATION.
0023 1 *
0024 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0025 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0026 1 *
0027 1 *
0028 1 *****
0029 1
0030 1
0031 1 **
0032 1 FACILITY:      DSR (Digital Standard RUNOFF) / DSRPLUS
0033 1
0034 1 ABSTRACT:      Processes .FIGURE and .FIGURE DEFERRED commands.
0035 1
0036 1 ENVIRONMENT:   Transportable
0037 1
0038 1 AUTHCR:        R.W.Friday      CREATION DATE:  May, 1978
0039 1

```

Revision History

```
.. 41 0040 1 %SBTTL 'Revision History'  
.. 42 0041 1   MODIFIED BY:  
.. 43 0042 1  
.. 44 0043 1   010  RER00010  Ron Randall  30-May-1983  
.. 45 0044 1   Changed maximum number of lines user can leave for a figure  
.. 46 0045 1   or deferred figure from logical page length to logical page  
.. 47 0046 1   length minus any header lines and bottom-of-page lines as  
.. 48 0047 1   specified in a LAYOUT command.  
.. 49 0048 1  
.. 50 0049 1   009  RER00009  Ron Randall  17-Mar-1983  
.. 51 0050 1   For DSRPLUS:  
.. 52 0051 1   Added code to handle figures in topnotes.  
.. 53 0052 1  
.. 54 0053 1   008  RER00008  Ron Randall  07-Mar-1983  
.. 55 0054 1   Global edit of all modules. Updated module names, idents,  
.. 56 0055 1   copyright dates. Changed require files to BLISS library.  
.. 57 0056 1   --  
.. 58 0057 1
```

Module Level Declarations

```

60      0058 1 %SBTTL 'Module Level Declarations'
61      0059 1
62      0060 1 : TABLE OF CONTENTS:
63      0061 1
64      0062 1 FORWARD ROUTINE
65      0063 1   gfig      : NOVALUE;
66      0064 1
67      0065 1 : INCLUDE FILES:
68      0066 1
69      0067 1 LIBRARY 'NXPORT:XPORT';           ! XPORT Library
70      0068 1 REQUIRE 'REQ:RNODEF';           ! RUNOFF variant definitions
71      0199 1
72      U 0200 1 %IF DSRPLUS %THEN
73      U 0201 1 LIBRARY 'REQ:DPLLIB';           ! DSRPLUS BLISS Library
74      0202 1 %ELSE
75      0203 1 LIBRARY 'REQ:DSRLIB';           ! DSR BLISS Library
76      0204 1 %FI
77      0205 1
78      0206 1
79      0207 1 : EXTERNAL REFERENCES:
80      0208 1
81      0209 1 EXTERNAL
82      0210 1   fncf      : fncf_definition,
83      0211 1   gca      : gca_definition,
84      0212 1   hct      : hct_definition,
85      0213 1   ira      : FIXED STRING,
86      0214 1   mra      : REF FIXED STRING,
87      0215 1   numprm   : numprm_define,
88      0216 1   phan     : phan_definition,
89      0217 1   tsf      : tsf_definition;
90      0218 1
91      U 0219 1 %IF DSRPLUS %THEN
92      UU 0220 1 EXTERNAL
93      U 0221 1   topnot   : tn_definition;
94      0222 1 %FI
95      0223 1
96      0224 1 EXTERNAL LITERAL
97      0225 1   rintes    : UNSIGNED (8);
98      0226 1
99      0227 1 EXTERNAL LITERAL
100     0228 1   rnfinm;
101     0229 1
102     0230 1 EXTERNAL ROUTINE
103     0231 1   erma,
104     0232 1   gcpage,
105     0233 1   guskip,
106     0234 1   outcrg;
107     0235 1

```

```

109 0236 1 GLOBAL ROUTINE figure (handler_code) : NOVALUE =
110 0237 1
111 0238 1 ++
112 0239 1 FUNCTIONAL DESCRIPTION:
113 0240 1
114 0241 1     Processes .FIGURE and .FIGURE DEFERRED commands.
115 0242 1
116 0243 1 FORMAL PARAMETERS:
117 0244 1
118 0245 1     handler_code - Indicates which command should be processed.
119 0246 1
120 0247 1 IMPLICIT INPUTS:
121 0248 1
122 0249 1     numprm - Contains a number, as processed by GETNUM.
123 0250 1
124 0251 1 IMPLICIT OUTPUTS:     None
125 0252 1
126 0253 1 ROUTINE VALUE:
127 0254 1 COMPLETION CODES:     None
128 0255 1
129 0256 1 SIDE EFFECTS:         None
130 0257 1 --
131 0258 1
132 0259 2     BEGIN
133 0260 2
134 0261 2     |
135 0262 2     | Illegal number. Ignore the command.
136 0263 2     |
137 0264 2     | IF NOT .num_result
138 0265 2     | THEN
139 0266 2     |     RETURN;
140 0267 2     |
141 0268 2     |
142 0269 2     | Default setting is 1.
143 0270 2     |
144 0271 2     | IF .num_length EQL 0
145 0272 2     | THEN
146 0273 2     |     num_value = 1;
147 0274 2     |
148 0275 2     |
149 0276 2     | Maximum space that can be left for a figure is "real" space on a page.
150 0277 2     |
151 0278 2     | IF (.num_value LEQ 0) OR
152 0279 4     |     (.num_value GTR (.phan_llines - .hct_layoutn
153 0280 5     |         - (IF .hct_subtitle THEN 1 ELSE 0)
154 0281 3     |         - (IF .hct_headers THEN 3 ELSE 0)))
155 0282 2     | THEN
156 0283 3     |     BEGIN
157 0284 3     |     erma (rnfinm, false);
158 0285 3     |     RETURN;
159 0286 2     |     END;
160 0287 2
161 0288 2     SELECTONE .handler_code OF
162 0289 2     SET
163 0290 2     [h_figure] : gfig (true, .num_value);
164 0291 2     [h_figure_deferr] : gfig (false, .num_value);
165 0292 2     TES;

```


FIGURE
V04-000
: 168

Module Level Declarations
0295 1

1 5
16-Sep-1984 00:27:33
14-Sep-1984 13:06:15

\\A-11 Bliss-32 V4.0-742
[RUNOFF.SRC]FIGURE.BLI;1

Page 6
(4)

**F

Module Level Declarations

```

170 0296 1 GLOBAL ROUTINE gfig (immediately, count) : NOVALUE =
171 0297 1
172 0298 1 ++
173 0299 1 FUNCTIONAL DESCRIPTION:
174 0300 1
175 0301 1     Generates code corresponding to RUNOFF's .FIGURE and
176 0302 1     .FIGURE DEFERRED commands.
177 0303 1
178 0304 1 FORMAL PARAMETERS:
179 0305 1
180 0306 1     immediately - Indicates whether the block of lines should be done
181 0307 1     immediately, or whether it can be put off until the
182 0308 1     top of the next page. TRUE means immediately.
183 0309 1
184 0310 1     count - Number of lines to be generated.
185 0311 1
186 0312 1 IMPLICIT INPUTS: None
187 0313 1
188 0314 1 IMPLICIT OUTPUTS: None
189 0315 1
190 0316 1 ROUTINE VALUE:
191 0317 1 COMPLETION CODES: None
192 0318 1
193 0319 1 SIDE EFFECTS: None
194 0320 1 --
195 0321 1
196 0322 2 BEGIN
197 0323 2 LOCAL
198 0324 2     tsf_phregs : REF VECTOR [tsf_nregs];
199 0325 2
200 0326 2     tsf_phregs = tsf__phregs;
201 0327 2
202 0328 2     |
203 0329 2     | If collecting a footnote, all figures are unconditional and cannot
204 0330 2     | start a new line. Code generation is particularly simple, being
205 0331 2     | just an unconditional generation of the number of lines.
206 0332 2     |
207 0333 2     IF .fnct_collecting
208 0334 2     THEN
209 0335 2         BEGIN
210 0336 2             guskip (.count);
211 0337 2             RETURN;
212 0338 2             END;
213 0339 2
214 U 0340 2 %IF DSRPLUS %THEN
215 UU 0341 2     |
216 UU 0342 2     | If collecting a topnote, all figures are unconditional.
217 UU 0343 2     |
218 UU 0344 2     IF .tn_collecting
219 UU 0345 2     THEN
220 UU 0346 2         BEGIN
221 UU 0347 2             guskip (.count);
222 UU 0348 2             RETURN;
223 UU 0349 2             END;
224 U 0350 2 %FI
225 0351 2
226 0352 2     IF .tsf_next_reg GE0 tsf_nregs ! Don't allocate too many "registers".

```

```

227 0353 2 THEN
228 0354 2     outcrg ();
229 0355 2
230 0356 2     tsf_phregs [.tsf_next_reg] = .count;
231 0357 2     fs_wchar (mra, rintes);           ! If (test page..)
232 0358 2     fs_wchar (mra, %C't');
233 0359 2     fs_wchar (mra, .tsf_next_reg);
234 0360 2     fs_wchar (mra, rintes);           ! then skip now
235 0361 2     fs_wchar (mra, %C'u');
236 0362 2     fs_wchar (mra, .tsf_next_reg);
237 0363 2     fs_wchar (mra, rintes);           ! end then
238 0364 2     fs_wchar (mra, %C'.');
239 0365 2     fs_wchar (mra, %C' ');
240 0366 2     tsf_int_vl = .tsf_int_vl + 9;
241 0367 2
242 0368 2 IF .immediately
243 0369 2 THEN
244 0370 2     BEGIN
245 0371 2     gcpage ();           ! Generate code for a new page.
246 0372 2     fs_wchar (mra, rintes);       ! generate code for skips
247 0373 2     fs_wchar (mra, %C'u');       ! unconditional skipping!
248 0374 2     fs_wchar (mra, .tsf_next_reg);
249 0375 2     END
250 0376 2 ELSE
251 0377 2     BEGIN           ! The block of lines can wait.
252 0378 2     fs_wchar (mra, rintes);
253 0379 2     fs_wchar (mra, %C'd');       ! defer space
254 0380 2     fs_wchar (mra, .tsf_next_reg);
255 0381 2     END;
256 0382 2
257 0383 2     fs_wchar (mra, rintes);           ! end else
258 0384 2     fs_wchar (mra, %C'.');
259 0385 2     fs_wchar (mra, %C' ');
260 0386 2     tsf_int_vl = .tsf_int_vl + 6;
261 0387 2     tsf_next_reg = .tsf_next_reg + 1;
262 0388 2     END;           ! End of GFIG

```

			00FC 0000	.ENTRY	GFIG, Save R2,R3,R4,R5,R6,R7	: 0296
	57	00G	8F 9A 000U2	MOVZBL	#RINTES, R7	:
	56	00000000G	EF 9E 00006	MOVAB	MRA, R6	:
	55	00000000G	EF 9E 00000	MOVAB	TSF, R5	:
52	65	0000008C	8F C1 00014	ADDL3	#140, TSF, TSF_PHREGS	: 0326
	08	00000000G	EF E9 0001C	BLBC	FNCT+20, f\$: 0333
		08	AC DD 00C23	PUSHL	COUNT	: 0336
	00000000G	EF	01 FB 00026	CALLS	#1, GUSKIP	:
			04 0002D	RET		: 0335
	50		65 D0 0002E 1\$:	MOVL	TSF, R0	: 0352
	05	0088	C0 D1 00031	CMPL	136(R0), #5	:
			07 19 00036	BLSS	2\$:
	00000000G	EF	00 FB 00038	CALLS	#0, OUTCRG	: 0354
	54		65 D0 0003F 2\$:	MOVL	TSF, R4	: 0356
	51	0088	C4 D0 00042	MOVL	136(R4), R1	:
	6241	08	AC D0 00047	MOVL	COUNT, (TSF_PHREGS)[R1]	:

Module Level Declarations

	50		66	D0	0004C	MOVL	MRA, R0	0357
	52	04	A0	9E	0004F	MOVAB	4(R0), R2	
00	B2		57	90	00053	MOVB	R7, @0(R2)	
	53	0C	A0	9E	00059	INCL	(R2)	
			62	D6	00057	MOVAB	12(R0), R3	
00	B2	74	63	D6	0005D	INCL	(R3)	0358
			8F	90	0005F	MOVB	#116, @0(R2)	
			62	D6	00064	INCL	(R2)	
00	B2		63	D6	00066	INCL	(R3)	0359
			51	90	00068	MOVB	R1, @0(R2)	
			62	D6	0006C	INCL	(R2)	
00	B2		63	D6	0006E	INCL	(R3)	0360
			57	90	00070	MOVB	R7, @0(R2)	
			62	D6	00074	INCL	(R2)	
00	B2	75	63	D6	00076	INCL	(R3)	0361
			8F	90	00078	MOVB	#117, @0(R2)	
			62	D6	0007D	INCL	(R2)	
00	B2		63	D6	0007F	INCL	(R3)	0362
			51	90	00081	MOVB	R1, @0(R2)	
			62	D6	00085	INCL	(R2)	
00	B2		63	D6	00087	INCL	(R3)	0363
			57	90	00089	MOVB	R7, @0(R2)	
			62	D6	0008D	INCL	(R2)	
00	B2		63	D6	00C8F	INCL	(R3)	0364
			2E	90	00091	MOVB	#46, @0(R2)	
			62	D6	00095	INCL	(R2)	
00	B2		63	D6	00097	INCL	(R3)	0365
			20	90	00099	MOVB	#32, @0(R2)	
			62	D6	0009D	INCL	(R2)	
18	A4		63	D6	0009F	INCL	(R3)	0366
	31	04	09	C0	000A1	ADDL2	#9, 24(R4)	0368
00000000G	EF		AC	E9	000A5	BLBC	IMMEDIATELY, 3\$	0371
	50		00	FB	000A9	CALLS	#0, GCPAGE	0372
	51	04	66	D0	000B0	MOVL	MRA, R0	
00	B1		A0	9E	000B3	MOVAB	4(R0), R1	
			57	90	000B7	MOVB	R7, @0(R1)	
		0C	61	D6	000BB	INCL	(R1)	
00	B1	75	A0	D6	000BD	INCL	12(R0)	0373
			8F	90	000C0	MOVB	#117, @0(R1)	
		0C	61	D6	000C5	INCL	(R1)	
	52		A0	D6	000C7	INCL	12(R0)	0374
00	B1	0088	65	D0	000CA	MOVL	TSF, R2	
			C2	90	000CD	MOVB	136(R2), @0(R1)	
		0C	61	D6	000D3	INCL	(R1)	
			A0	D6	000D5	INCL	12(R0)	
00	B2		1B	11	000D8	BRB	4\$	0368
			57	90	000DA	MOVB	R7, @0(R2)	0378
			62	D6	000DE	INCL	(R2)	
00	B2	64	63	D6	000E0	INCL	(R3)	0379
			8F	90	000E2	MOVB	#100, @0(R2)	
			62	D6	000E7	INCL	(R2)	
00	B2	0088	63	D6	000E9	INCL	(R3)	0380
			C4	90	000EB	MOVB	136(R4), @0(R2)	
			62	D6	000F1	INCL	(R2)	
			63	D6	000F3	INCL	(R3)	
	50		66	D0	000F5	MOVL	MRA, R0	0383
	51	04	A0	9E	000F8	MOVAB	4(R0), R1	

```

00 B1          57 90 000FC      MOVB   R7, @0(R1)
                61 D6 00100      INCL   (R1)
                0C A0 D6 00102      INCL  12(R0)
00 B1          2E 90 00105      MOVB  #46, @0(R1)
                61 D6 00109      INCL   (R1)
                0C A0 D6 0010B      INCL  12(R0)
00 B1          20 90 0010E      MOVB  #32, @0(R1)
                61 D6 00112      INCL   (R1)
                0C A0 D6 00114      INCL  12(R0)
                50 65 D0 00117      MOVL  TSF, R0
18 A0          06 C0 0011A      ADDL2 #6, 24(R0)
                0088 C0 D6 0011E      INCL  136(R0)
                04 00122      RET
    
```

```

:
:
: 0384
:
: 0385
:
:
: 0386
: 0387
: 0388
    
```

: Routine Size: 291 bytes, Routine Base: \$CODE\$ + 0082

```

: 263          0389 1
: 264          0390 1 END
: 265          0391 0 ELUDOM
    
```

! End of module

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	421	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	00:00.1
\$255\$DUA28:[RUNOFF.SRC]DSRLIB.L32;1	1248	32	2	86	00:00.3

COMMAND QUALIFIERS

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

```

: Size:          421 code + 0 data bytes
: Run Time:      00:10.3
: Elapsed Time: 00:34.3
: Lines/CPU Min: 2277
    
```


