

| | | | |
|--------------|-----|------------|----------------|
| DDDDDDDDDDDD | | IIIIIIIIII | FFFFFFFFFFFFFF |
| DDDDDDDDDDDD | | IIIIIIIIII | FFFFFFFFFFFFFF |
| DDDDDDDDDDDD | | IIIIIIIIII | FFFFFFFFFFFFFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFFFFFFFFFFFFF |
| DDD | DDD | III | FFFFFFFFFFFFFF |
| DDD | DDD | III | FFFFFFFFFFFFFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDD | DDD | III | FFF |
| DDDDDDDDDDDD | | IIIIIIIIII | FFF |
| DDDDDDDDDDDD | | IIIIIIIIII | FFF |
| DDDDDDDDDDDD | | IIIIIIIIII | FFF |



```

DDDDDDDD      IIIIII      FFFFFFFFFF      GGGGGGGG      EEEEEEEEEE      TTTTTTTTTT      CCCCCCCC      MM      MM      DDDDDDDD
DDDDDDDD      IIIIII      FFFFFFFFFF      GGGGGGGG      EEEEEEEEEE      TTTTTTTTTT      CCCCCCCC      MM      MM      DDDDDDDD
DD      DD      II      FF      GG      EE      TT      CC      MMMM      MMMM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MMMM      MMMM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MM      MM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MM      MM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MM      MM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MM      MM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MM      MM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MM      MM      DD      DD
DD      DD      II      FF      GG      EE      TT      CC      MM      MM      DD      DD
DDDDDDDD      IIIIII      FFFFFFFFFF      GGGGGGGG      EEEEEEEEEE      TTTTTTTTTT      CCCCCCCC      MM      MM      DDDDDDDD
DDDDDDDD      IIIIII      FFFFFFFFFF      GGGGGGGG      EEEEEEEEEE      TTTTTTTTTT      CCCCCCCC      MM      MM      DDDDDDDD

```

```

LL      LL      IIIIII      SSSSSSSS
LL      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
LLLLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLLLL      IIIIII      SSSSSSSS

```

```

1 0001 0 MODULE DIF_GETCMD ( ! Differences command processing
2 0002 0 LANGUAGE (BLISS32),
3 0003 0 ADDRESSING_MODE(EXTERNAL=GENERAL,
4 0004 0 NONEXTERNAL=LONG_RELATIVE),
5 0005 0 IDENT = 'V04-000'
6 0006 0 ) =
7 0007 1 BEGIN
8 0008 1 .....
9 0009 1 *
10 0010 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
11 0011 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
12 0012 1 * ALL RIGHTS RESERVED. *
13 0013 1 *
14 0014 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
15 0015 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
16 0016 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
17 0017 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
18 0018 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
19 0019 1 * TRANSFERRED. *
20 0020 1 *
21 0021 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
22 0022 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
23 0023 1 * CORPORATION. *
24 0024 1 *
25 0025 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
26 0026 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
27 0027 1 *
28 0028 1 *
29 0029 1 .....
30 0030 1
31 0031 1 **
32 0032 1
33 0033 1 FACILITY: DCL Differences command
34 0034 1
35 0035 1 ABSTRACT:
36 0036 1 The DCL DIFFERENCES command compares the contents of
37 0037 1 two files.
38 0038 1
39 0039 1 ENVIRONMENT:
40 0040 1 VAX native, user mode
41 0041 1
42 0042 1 --
43 0043 1
44 0044 1 AUTHOR: Peter George, Benn Schreiber CREATION DATE: 1-August-1981
45 0045 1
46 0046 1 MODIFIED BY:
47 0047 1
48 0048 1 V03-003 PCG0007 Peter George 25-Apr-1983
49 0049 1 Set /NONUMBER when /SLP is specified.
50 0050 1
51 0051 1 V03-002 PCG0006 Peter George 30-Mar-1983
52 0052 1 Use CLI keyword parsing.
53 0053 1
54 0054 1 V03-001 PCG0005 Peter George 11-Aug-1982
55 0055 1 Remove reference to CLISEND_PARSE.
56 0056 1 --
57 0057 1

```

Declarations

```

59 0058 1 %SBTTL 'Declarations';
60 0059 1 LIBRARY
61 0060 1 'SYSSLIBRARY:STARLET.L32'; ! System macros
62 0061 1 LIBRARY
63 0062 1 'SYSSLIBRARY:TPAMAC.L32'; ! TPARSE macros
64 0063 1 REQUIRE
65 0064 1 'DIFPRE'; ! DIF prefix file
66 0138 1 REQUIRE
67 0139 1 'DIFDEF'; ! DIF data structures
68 0370 1
69 0371 1 EXTERNAL
70 0372 1 dif$gl_typedata : BBLOCK, ! File type to comment delimiter info
71 0373 1 dif$gl_commdesc : BBLOCK, ! Desc for buffer of comment delimiters
72 0374 1 dif$gl_commflgs : BITVECTOR, ! Bit flags for comment chars in col. 1 only
73 0375 1 dif$gl_ignore : BBLOCK, ! Flags of characters to ignore
74 0376 1 dif$gl_cmddesc : BBLOCK, ! Command line descriptor
75 0377 1 dif$gl_header, ! No. of lines to skip as header
76 0378 1 dif$gl_width, ! Width of lines in output listing
77 0379 1 dif$gl_match, ! No. of records that constitute a match
78 0380 1 dif$gl_maxdif, ! Maximum number of unmatched records
79 0381 1 dif$gl_merged, ! No. of matched lines to follow each list of differences
80 0382 1 dif$gl_parallel, ! Same as above for parallel
81 0383 1 dif$gl_wndwsiz, ! No. of records to search before declaring a mismatch
82 0384 1 dif$gl_flags : BBLOCK, ! Flags
83 0385 1
84 0386 1 ! Input and output file data structures
85 0387 1
86 0388 1 dif$gl_masfdb : BBLOCK, ! Master file fdb
87 0389 1 dif$gl_masdesc : BBLOCK, ! String descriptor for input file
88 0390 1 dif$gl_masrab : BBLOCK, ! RAB for master file
89 0391 1 dif$gl_masnam : REF BBLOCK, ! pointer to NAM block for master file
90 0392 1 dif$gl_masrss : BBLOCK, ! Resultant name string for master file
91 0393 1 dif$gl_revfdb : BBLOCK, ! Revision file fdb
92 0394 1 dif$gl_revdesc : BBLOCK, ! String descriptor for revision file
93 0395 1 dif$gl_revrab : BBLOCK, ! RAB for revision file
94 0396 1 dif$gl_revnam : BBLOCK, ! NAM block for revision file
95 0397 1 dif$gl_revrss : BBLOCK, ! Resultant name string for revision file
96 0398 1 dif$gl_outifi, ! IFI of output file
97 0399 1 dif$gl_outdesc : BBLOCK, ! String descriptor for output file
98 0400 1 dif$gl_outrab : BBLOCK, ! RAB for output file
99 0401 1 dif$gl_outnam : BBLOCK, ! NAM block for output file
100 0402 1 dif$gl_outrss : BBLOCK; ! Output file resultant name string
101 0403 1
102 0404 1 EXTERNAL ROUTINE
103 0405 1 cli$gl_value, ! Get qualifier value
104 0406 1 cli$gl_present, ! Test if qualifier present
105 0407 1 lib$cvd_dtb, ! Convert decimal to binary
106 0408 1 lib$find_file, ! Wild-card parse/search
107 0409 1 lib$gl_vm, ! Allocate virtual memory
108 0410 1 lib$scopy_r_dx, ! Copy a string
109 0411 1 lib$tparse, ! Table driven parser
110 0412 1 str$copy_dx, ! Copy string using descriptors
111 0413 1 SYSSFAD; ! Formatted ASCII output conversion
112 0414 1
113 0415 1 EXTERNAL LITERAL
114 0416 1 cli$gl_present, ! Qualifier explicitly present
115 0417 1 cli$gl_negated, ! Qualifier explicitly negated

```

```

: 116      0418 1      dif$c_maxlangs,      ! Number of languages in dif$gl_typedata table
: 117      0419 1      dif$_badvalue,      ! Bad qualifier value
: 118      0420 1      dif$_closein,      ! Error closing input file
: 119      0421 1      dif$_closeout,      ! Error closing output file
: 120      0422 1      dif$_cmdlin,      ! To print bad command line
: 121      0423 1      dif$_cmdsyntax,      ! Command syntax error
: 122      0424 1      dif$_confqual,      ! Conflicting qualifiers
: 123      0425 1      dif$_mrggtmtch,      ! /MERGE greater than /MATCH
: 124      0426 1      dif$_nolower,      ! No lower version of file found
: 125      0427 1      dif$_novalue,      ! Qualifier requires a value
: 126      0428 1      dif$_nowild,      ! No wild cards allowed
: 127      0429 1      dif$_openin,      ! Error opening input file
: 128      0430 1      dif$_openout,      ! Error opening output file
: 129      0431 1      dif$_pargtmtch,      ! /PARALLEL greater than /MATCH
: 130      0432 1      dif$_parsefail,      ! Error from parse
: 131      0433 1      dif$_strcpyerr,      ! Error from string copy
: 132      0434 1      dif$_sysfacerr,      ! Error from fao
: 133      0435 1      dif$_toomnydlm,      ! Too many comment delimiters
: 134      0436 1      dif$_toomnyfil,      ! Too many input files
: 135      0437 1      dif$_twowidths;      ! Both /LINE_WIDTH and /WIDTH specified
: 136      0438 1
: 137      0439 1      FORWARD ROUTINE
: 138      0440 1      call_tparse,      ! Call TPARSE
: 139      0441 1      syntaxerr,      ! Print syntax error message
: 140      0442 1      badvaluerr,      ! Illegal qualifier value
: 141      0443 1      getchbarqual,      ! Get /change_bar qualifier
: 142      0444 1      getfilename,      ! Get file spec from FAB/NAM
: 143      0445 1      initfile,      ! Init RMS data to open a file
: 144      0446 1      dif$open_mas,      ! Open master input file
: 145      0447 1      dif$open_rev,      ! Open the revision file
: 146      0448 1      dif$open_out,      ! Open the output file
: 147      0449 1      dif$close_in,      ! Close an input file
: 148      0450 1      dif$close_out,      ! Close the output file
: 149      0451 1      allocate_ubf,      ! Allocate user buffer for records that cross block boundari
: 150      0452 1      find_list_width;      ! Find width of listing device if /WIDTH not specified
: 151      0453 1
: 152      0454 1      OWN
: 153      0455 1      sys$output : COUNTEDSTRING ('SYS$OUTPUT'),      ! Default output file
: 154      0456 1      tpalindesc : REF BBLOCK,      ! Pointer to current string descr
: 155      0457 1      tpdata,      ! Data returned from TPARSE
: 156      0458 1      tpadding : BBLOCK[tpa$length0],      ! Tparse block
: 157      0459 1      find_context : REF BBLOCK,      ! Context for lib$find_file
: 158      0460 1      find_masfile : desc decl      ! String descriptor for master input file
: 159      0461 1      PRESET T[dsc$b_class] = dsc$b_class_d);
: 160      0462 1
: 161      0463 1      LITERAL
: 162      0464 1      ffeed = 12;      ! Formfeed character
: 163      0465 1      !
: 164      0466 1      ! Declare static strings for qualifier names
: 165      0467 1      !
: 166      0468 1      MACRO
: 167      M 0469 1      SD[A] =
: 168      0470 1      BIND %NAME('SD_',A) = $descriptor(a)%;
: 169      0471 1
: 170      P 0472 1      SD(
: 171      P 0473 1      '$LINE',
: 172      P 0474 1      'INPUT1',

```

Declarations

```
.. 173 P 0475 1 'INPUT2'  
.. 174 P 0476 1 'CHANGE BAR'  
.. 175 P 0477 1 'COMMENT_DELIMITERS',  
.. 176 P 0478 1 'IGNORE',  
.. 177 P 0479 1 'LINE_WIDTH',  
.. 178 P 0480 1 'MATCH',  
.. 179 P 0481 1 'MAXIMUM_DIFFERENCES',  
.. 180 P 0482 1 'MERGED',  
.. 181 P 0483 1 'MODE',  
.. 182 P 0484 1 'NUMBER',  
.. 183 P 0485 1 'OUTPUT',  
.. 184 P 0486 1 'PARALLEL',  
.. 185 P 0487 1 'SEPARATED',  
.. 186 P 0488 1 'SLP',  
.. 187 P 0489 1 'WIDTH',  
.. 188 P 0490 1 'WINDOW',  
.. 189 P 0491 1 'BLANK_LINES',  
.. 190 P 0492 1 'COMMENTS',  
.. 191 P 0493 1 'FORM_FEEDS',  
.. 192 P 0494 1 'HEADER',  
.. 193 P 0495 1 'TRAILING_SPACES',  
.. 194 P 0496 1 'SPACING',  
.. 195 P 0497 1 'EXACT',  
.. 196 P 0498 1 'PRETTY',  
.. 197 P 0499 1 'ASCII',  
.. 198 P 0500 1 'HEXADECIMAL',  
.. 199 P 0501 1 'OCTAL',  
.. 200 P 0502 1 'MASTER',  
.. 201 P 0503 1 'REVISION');
```

```
TPARSE state tables
: 203 0504 1 %SBTTL 'TPARSE state tables';
: 204 0505 1
: 205 0506 1
: 206 0507 1 !! TPARSE tables to parse /COMMENT_DELIMITERS qualifier values
: 207 0508 1
: 208 0509 1 $INIT_STATE(comment_delim_states,delim_keys);
: 209 0510 1
: 210 P 0511 1 $STATE(,
: 211 P P 0512 1 ('COLON',tpa$_exit,,':',tpdata),
: 212 P P 0513 1 ('COMMA',tpa$_exit,,',',tpdata),
: 213 P P 0514 1 ('EXCLAMATION',tpa$_exit,,':!',tpdata),
: 214 P P 0515 1 ('FORM FEED',tpa$_exit,,f'feed',tpdata),
: 215 P P 0516 1 ('LEFT',tpa$_exit,,'[',tpdata),
: 216 P P 0517 1 ('RIGHT',tpa$_exit,,']',tpdata),
: 217 P P 0518 1 ('SEMI COLON',tpa$_exit,,':',tpdata),
: 218 P P 0519 1 ('SLASH',tpa$_exit,, '/',tpdata),
: 219 P P 0520 1 ('SPACE',tpa$_exit,, ' ',tpdata),
: 220 P P 0521 1 ('TAB',tpa$_exit,, ' ',tpdata),
: 221 P 0522 1 (tpa$_(lambda,,syntaxerr)
: 222 0523 1 );
: 223 0524 1
: 224 0525 1 !! TPARSE tables to parse /CHANGE_BAR qualifier value
: 225 0526 1
: 226 0527 1
: 227 0528 1 $INIT_STATE(linenum_states,linenum_keys);
: 228 0529 1
: 229 P 0530 1 $STATE(,
: 230 P P 0531 1 ('NUMBER',tpa$_exit,,1,tpdata),
: 231 P 0532 1 ('NONNUMBER',tpa$_exit)
: 232 0533 1 );
```

TPARSE state tables

```

234 0534 1 ROUTINE call_tparse(string,states,keys) =
235 0535 2 BEGIN
236 0536 2
237 0537 2
238 0538 2 This routine calls tparse given the string, states and keys
239 0539 2
240 0540 2 Inputs:
241 0541 2
242 0542 2 string address of string descriptor for string to parse
243 0543 2 states address of tparse states table
244 0544 2 keys address of tparse keys table
245 0545 2
246 0546 2
247 0547 2 MAP
248 0548 2 string : REF BBLOCK;
249 0549 2
250 0550 2 LOCAL
251 0551 2 status;
252 0552 2
253 0553 2
254 0554 2 Initialize the tparse block
255 0555 2
256 0556 2 CHSFILL(0,tpaSk_length0,tpablock);
257 0557 2 tpablock[tpa$l_count] = tpaSk_count0;
258 0558 2 tpablock[tpa$l_options] = tpa$m_abbrev;
259 0559 2 tpablock[tpa$l_stringcnt] = .string[dsc$w_length];
260 0560 2 tpablock[tpa$l_stringptr] = .string[dsc$a_pointer];
261 0561 2
262 0562 2 Initialize OWN storage used
263 0563 2
264 0564 2 tpalindesc = .string;
265 0565 2 tpdata = 0;
266 0566 2
267 0567 2 RETURN lib$tparse(tpablock,.states,.keys)
268 0568 2
269 0569 1 END;

```

```

.TITLE DIF_GETCMD
.IDENT \V04-000\
.PSECT _LIB$KEY1$,NOWRT, SHR, PIC,1

```

```

00000 :TPASKEYSTO
U.2: .BLKB 0
4E 4F 4C 4F 43 00000 :TPASKEYST
U.4: .ASCII \COLON\
FF 00005 :TPASKEYST
: .BYTE -1
00006 :TPASKEYSTO
U.9: .BLKB 0
41 4D 4D 4F 43 00006 :TPASKEYST
U.11: .ASCII \COMMA\
FF 0000B :TPASKEYST
: .BYTE -1
0000C :TPASKEYSTO
U.16: .BLKB 0
4E 4F 49 54 41 4D 41 4C 43 58 45 0000C :TPASKEYST
U.18: .ASCII \EXCLAMATION\

```



```
FF 00017 .BYTE -1
00018 ;TPASKEYSTO
U.23: .BLKB 0
44 45 45 46 5F 4D 52 4F 46 00018 ;TPASKEYST
U.25: .ASCII \FORM_FEED\
FF 00021 .BYTE -1
00022 ;TPASKEYSTO
U.30: .BLKB 0
54 46 45 4C 00022 ;TPASKEYST
U.32: .ASCII \LEFT\
FF 00026 .BYTE -1
00027 ;TPASKEYSTO
U.37: .BLKB 0
54 48 47 49 52 00027 ;TPASKEYST
U.39: .ASCII \RIGHT\
FF 0002C .BYTE -1
0002D ;TPASKEYSTO
U.44: .BLKB 0
4E 4F 4C 4F 43 5F 49 4D 45 53 0002D ;TPASKEYST
U.46: .ASCII \SEMI_COLON\
FF 00037 .BYTE -1
00038 ;TPASKEYSTO
U.51: .BLKB 0
48 53 41 4C 53 00038 ;TPASKEYST
U.53: .ASCII \SLASH\
FF 0003D .BYTE -1
0003E ;TPASKEYSTO
U.58: .BLKB 0
45 43 41 50 53 0003E ;TPASKEYST
U.60: .ASCII \SPACE\
FF 00043 .BYTE -1
00044 ;TPASKEYSTO
U.65: .BLKB 0
42 41 54 00044 ;TPASKEYST
U.67: .ASCII \TAB\
FF 00047 .BYTE -1
FF 00048 ;TPASKEYFILL
U.74: .BYTE -1
00049 ;TPASKEYSTO
U.76: .BLKB 0
52 45 42 4D 55 4E 00049 ;TPASKEYST
U.78: .ASCII \NUMBER\
FF 0004F .BYTE -1
00050 ;TPASKEYSTO
U.83: .BLKB 0
52 45 42 4D 55 4E 4F 4E 00050 ;TPASKEYST
U.85: .ASCII \NONUMBER\
FF 00058 .BYTE -1
FF 00059 ;TPASKEYFILL
U.88: .BYTE -1

.PSECT _LIB$STATES,NOWRT, SHR, PIC,1
0000 COMMENT_DELIM_STATES::
.BLKB 0
7100 0000 ;TPATYPE
U.5: .WORD 28928
```

.....

```

00000000* 00002 ;TPASADDR
00 00 00 3A 00006 U.6: .LONG <<TPDATA-U.6>-4> :
          ;TPASMASK U.7: .ASCII \:\<0><0><0> :
          FFFF 0000A ;TPASTARGET U.8: .WORD -1 :
          7101 0000C ;TPASTYPE U.12: .WORD 28929 :
00000000* 0000E ;TPASADDR U.13: .LONG <<TPDATA-U.13>-4> :
00 00 00 2C 00012 ;TPASMASK U.14: .ASCII \,\<0><0><0> :
          FFFF 00016 ;TPASTARGET U.15: .WORD -1 :
          7102 00018 ;TPASTYPE U.19: .WORD 28930 :
00000000* 0001A ;TPASADDR U.20: .LONG <<TPDATA-U.20>-4> :
00 00 00 21 0001E ;TPASMASK U.21: .ASCII \!\<0><0><0> :
          FFFF 00022 ;TPASTARGET U.22: .WORD -1 :
          7103 00024 ;TPASTYPE U.26: .WORD 28931 :
00000000* 00026 ;TPASADDR U.27: .LONG <<TPDATA-U.27>-4> :
0000000C 0002A ;TPASMASK U.28: .LONG 12 :
          FFFF 0002E ;TPASTARGET U.29: .WORD -1 :
          7104 00030 ;TPASTYPE U.33: .WORD 28932 :
00000000* 00032 ;TPASADDR U.34: .LONG <<TPDATA-U.34>-4> :
00 00 00 5B 00036 ;TPASMASK U.35: .ASCII \[\<0><0><0> :
          FFFF 0003A ;TPASTARGET U.36: .WORD -1 :
          7105 0003C ;TPASTYPE U.40: .WORD 28933 :
00000000* 0003E ;TPASADDR U.41: .LONG <<TPDATA-U.41>-4> :
00 00 00 5D 00042 ;TPASMASK U.42: .ASCII \]\<0><0><0> :
          FFFF 00046 ;TPASTARGET U.43: .WORD -1 :
          7106 00048 ;TPASTYPE U.47: .WORD 28934 :
00000000* 0004A ;TPASADDR U.48: .LONG <<TPDATA-U.48>-4> :
00 00 00 3B 0004E ;TPASMASK U.49: .ASCII \;\<0><0><0> :
          FFFF 00052 ;TPASTARGET U.50: .WORD -1 :
          7107 00054 ;TPASTYPE U.54: .WORD 28935 :
00000000* 00056 ;TPASADDR

```

| | | | | | | | |
|----|----|-----------|----|-------|------------------|-------------------------------|---|
| 00 | 00 | 00 | 2F | 0005A | U.55: .LONG | <<TPDATA-U.55>-4> | : |
| | | | | | :TPASMASK | | : |
| | | FFFF | | 0005E | U.56: .ASCII | \ \<0><0><0> | : |
| | | | | | :TPASTARGET | | : |
| | | 7108 | | 00060 | U.57: .WORD | -1 | : |
| | | | | | :TPASTYPE | | : |
| | | 00000000* | | 00062 | U.61: .WORD | 28936 | : |
| | | | | | :TPASADDR | | : |
| 00 | 00 | 00 | 20 | 00066 | U.62: .LONG | <<TPDATA-U.62>-4> | : |
| | | | | | :TPASMASK | | : |
| | | FFFF | | 0006A | U.63: .ASCII | \ \<0><0><0> | : |
| | | | | | :TPASTARGET | | : |
| | | 7109 | | 0006C | U.64: .WORD | -1 | : |
| | | | | | :TPASTYPE | | : |
| | | 00000000* | | 0006E | U.68: .WORD | 28937 | : |
| | | | | | :TPASADDR | | : |
| 00 | 00 | 00 | 09 | 00072 | U.69: .LONG | <<TPDATA-U.69>-4> | : |
| | | | | | :TPASMASK | | : |
| | | FFFF | | 00076 | U.70: .ASCII | <9><0><0><0> | : |
| | | | | | :TPASTARGET | | : |
| | | 85F6 | | 00078 | U.71: .WORD | -1 | : |
| | | | | | :TPASTYPE | | : |
| | | 00000000V | | 0007A | U.72: .WORD | -31242 | : |
| | | | | | :TPASACTION | | : |
| | | 0007E | | | U.73: .LONG | <<SYNTAXERR-U.73>-4> | : |
| | | 00080 | | | .BLKB | 2 | : |
| | | | | | LINENUM_STATES:: | | : |
| | | 7100 | | 00080 | .BLKB | 0 | : |
| | | | | | :TPASTYPE | | : |
| | | 00000000* | | 00082 | U.79: .WORD | 28928 | : |
| | | | | | :TPASADDR | | : |
| | | 00000001 | | 00086 | U.80: .LONG | <<TPDATA-U.80>-4> | : |
| | | | | | :TPASMASK | | : |
| | | FFFF | | 0008A | U.81: .LONG | 1 | : |
| | | | | | :TPASTARGET | | : |
| | | 1501 | | 0008C | U.82: .WORD | -1 | : |
| | | | | | :TPASTYPE | | : |
| | | FFFF | | 0008E | U.86: .WORD | 5377 | : |
| | | | | | :TPASTARGET | | : |
| | | | | | U.87: .WORD | -1 | : |
| | | | | | .PSECT | _LIB\$KEYOS,NOWRT, SHR, PIC,1 | : |
| | | 00000 | | 00000 | DELIM_KEYS:: | | : |
| | | | | | .BLKB | 0 | : |
| | | 00000 | | 00000 | :TPASKEY0 | | : |
| | | 0000* | | 00000 | U.1: .BLKB | 0 | : |
| | | | | | :TPASKEY | | : |
| | | 0000* | | 00002 | U.3: .WORD | <U.2-U.1> | : |
| | | | | | :TPASKEY | | : |
| | | 0000* | | 00004 | U.10: .WORD | <U.9-U.1> | : |
| | | | | | :TPASKEY | | : |
| | | 0000* | | 00006 | U.17: .WORD | <U.16-U.1> | : |
| | | | | | :TPASKEY | | : |
| | | 0000* | | 00008 | U.24: .WORD | <U.23-U.1> | : |
| | | | | | :TPASKEY | | : |
| | | | | | U.31: .WORD | <U.30-U.1> | : |

```

0000* 0000A :TPASKEY
U.38: .WORD <U.37-U.1>
0000* 0000C :TPASKEY
U.45: .WORD <U.44-U.1>
0000* 0000E :TPASKEY
U.52: .WORD <U.51-U.1>
0000* 00010 :TPASKEY
U.59: .WORD <U.58-U.1>
0000* 00012 :TPASKEY
U.66: .WORD <U.65-U.1>
00014 LINENUM_KEYS::
      BLKB 0
00014 :TPASKEY
U.75: .BLKB 0
0000* 00014 :TPASKEY
U.77: .WORD <U.76-U.75>
0000* 00016 :TPASKEY
U.84: .WORD <U.83-U.75>

.PSECT SPLITS, NOWRT, NOEXE, 2

      45 4E 49 4C 24 00000 P.AAB: .ASCII \SLINE\
      00005 .BLKB 3
      00008 P.AAA: .LONG 5
      00000000' 0000C .ADDRESS P.AAB
31 54 55 50 4E 49 00010 P.AAD: .ASCII \INPUT1\
      00016 .BLKB 2
      00000006' 00018 P.AAC: .LONG 6
      00000000' 0001C .ADDRESS P.AAD
32 54 55 50 4E 49 00020 P.AAF: .ASCII \INPUT2\
      00026 .BLKB 2
      00000006' 00028 P.AAE: .LONG 6
      00000000' 0002C .ADDRESS P.AAF
      52 41 42 5F 45 47 4E 41 48 43 00030 P.AAH: .ASCII \CHANGE_BAR\
      0003A .BLKB 2
      0000000A' 0003C P.AAG: .LONG 10
      00000000' 00040 .ADDRESS P.AAH
54 49 4D 49 4C 45 44 5F 54 4E 45 4D 4D 4F 43 00044 P.AAJ: .ASCII \COMMENT_DELIMITERS\
      53 52 45 00053 .BLKB 2
      00000012' 00056 .LONG 18
      00000000' 00058 P.AAI: .ADDRESS P.AAJ
      45 52 4F 4E 47 49 00060 P.AAL: .ASCII \IGNORE\
      00066 .BLKB 2
      00000006' 00068 P.AAK: .LONG 6
      00000000' 0006C .ADDRESS P.AAL
      48 54 44 49 57 5F 45 4E 49 4C 00070 P.AAN: .ASCII \LINE_WIDTH\
      0007A .BLKB 2
      0000000A' 0007C P.AAM: .LONG 10
      00000000' 00080 .ADDRESS P.AAN
      48 43 54 41 4D 00084 P.AAP: .ASCII \MATCH\
      00089 .BLKB 3
      00000005' 0008C P.AAO: .LONG 5
      00000000' 00090 .ADDRESS P.AAP
45 52 45 46 46 49 44 5F 4D 55 4D 49 58 41 4D 00094 P.AAR: .ASCII \MAXIMUM_DIFFERENCES\
      53 45 43 4E 000A3 .BLKB 1
      000A7

```

| | | | | | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----------|-------|--------|----------|-------------------|--------|----------|--------|----------|--------------|--------|----------|-------|----|
| | | | | | | 00000013 | 000A8 | P.AAQ: | .LONG | 19 | | | | | | | | | |
| | | | | | | 0000000C | 000AC | | .ADDRESS | P.AAR | | | | | | | | | |
| 44 | 45 | 47 | 52 | 45 | 4D | 000B0 | 000B0 | P.AAT: | .ASCII | \MERGED\ | | | | | | | | | |
| | | | | | | | 000B6 | | .BLKB | 2 | | | | | | | | | |
| | | | | | | 00000006 | 000B8 | P.AAS: | .LONG | 6 | | | | | | | | | |
| | | | 45 | 44 | 4F | 4D | 000BC | | .ADDRESS | P.AAT | | | | | | | | | |
| | | | | | | 00000000 | 000C0 | P.AAV: | .ASCII | \MODE\ | | | | | | | | | |
| | | | | | | 00000004 | 000C4 | P.AAU: | .LONG | 4 | | | | | | | | | |
| | | | 52 | 45 | 42 | 4D | 55 | 4E | 000C8 | 000C8 | P.AAV: | .ADDRESS | P.AAV | | | | | | |
| | | | | | | | 000D2 | | .ASCII | \NUMBER\ | | | | | | | | | |
| | | | | | | 00000006 | 000D4 | P.AAX: | .BLKB | 2 | | | | | | | | | |
| | | | | | | 00000000 | 000D8 | P.AAW: | .LONG | 6 | | | | | | | | | |
| | | | 54 | 55 | 50 | 54 | 55 | 4F | 000DC | 000DC | P.AAZ: | .ADDRESS | P.AAX | | | | | | |
| | | | | | | | 000E2 | | .ASCII | \OUTPUT\ | | | | | | | | | |
| | | | | | | 00000006 | 000E4 | P.AAY: | .BLKB | 2 | | | | | | | | | |
| | | | | | | 00000000 | 000E8 | | .LONG | 6 | | | | | | | | | |
| 4C | 45 | 4C | 4C | 41 | 52 | 41 | 50 | | .ADDRESS | P.AAZ | | | | | | | | | |
| | | | | | | 00000008 | 000F4 | P.ABB: | .ASCII | \PARALLEL\ | | | | | | | | | |
| | | | | | | 00000000 | 000F8 | P.ABA: | .LONG | 8 | | | | | | | | | |
| 44 | 45 | 54 | 41 | 52 | 41 | 50 | 45 | 53 | 000FC | 000FC | P.ABB: | .ADDRESS | P.ABB | | | | | | |
| | | | | | | | 00105 | | .ASCII | \SEPARATED\ | | | | | | | | | |
| | | | | | | 00000009 | 00108 | P.ABC: | .BLKB | 3 | | | | | | | | | |
| | | | | | | 00000000 | 0010C | | .LONG | 9 | | | | | | | | | |
| | | | | | | 50 | 4C | 53 | 00110 | 00110 | P.ABD: | .ADDRESS | P.ABD | | | | | | |
| | | | | | | | 00113 | | .ASCII | \SLP\ | | | | | | | | | |
| | | | | | | 00000003 | 00114 | P.ABF: | .BLKB | 1 | | | | | | | | | |
| | | | | | | 00000000 | 00118 | P.ABE: | .LONG | 3 | | | | | | | | | |
| | | | 48 | 54 | 44 | 49 | 57 | | .ADDRESS | P.ABF | | | | | | | | | |
| | | | | | | | 00121 | P.ABH: | .ASCII | \WIDTH\ | | | | | | | | | |
| | | | | | | 00000005 | 00124 | P.ABG: | .BLKB | 3 | | | | | | | | | |
| | | | | | | 00000000 | 00128 | | .LONG | 5 | | | | | | | | | |
| | | | 57 | 4F | 44 | 4E | 49 | 57 | 0012C | 0012C | P.ABH: | .ADDRESS | P.ABH | | | | | | |
| | | | | | | | 00132 | | .ASCII | \WINDOW\ | | | | | | | | | |
| | | | | | | 00000006 | 00134 | P.ABJ: | .BLKB | 2 | | | | | | | | | |
| | | | | | | 00000000 | 00138 | | .LONG | 6 | | | | | | | | | |
| 53 | 45 | 4E | 49 | 4C | 5F | 4B | 4E | 41 | 4C | 42 | 0013C | 0013C | P.ABI: | .ADDRESS | P.ABJ | | | | |
| | | | | | | | | | .ASCII | \BLANK_LINES\ | | | | | | | | | |
| | | | | | | 0000000B | 00147 | | .BLKB | 1 | | | | | | | | | |
| | | | | | | 00000000 | 00148 | P.ABK: | .LONG | 11 | | | | | | | | | |
| | | | | | | 53 | 54 | 4E | 45 | 4D | 4D | 4F | 43 | 0014C | 0014C | P.ABL: | .ADDRESS | P.ABK | |
| | | | | | | | | | .ASCII | \COMMENTS\ | | | | | | | | | |
| | | | | | | 00000008 | 00150 | P.ABN: | .LONG | 8 | | | | | | | | | |
| | | | | | | 00000000 | 00158 | P.ABM: | .ADDRESS | P.ABN | | | | | | | | | |
| 53 | 44 | 45 | 45 | 46 | 5F | 4D | 52 | 4F | 46 | | 0015C | 0015C | P.ABN: | .ASCII | \FORM_FEEDS\ | | | | |
| | | | | | | | | | | | 00160 | 00160 | P.ABP: | .BLKB | 2 | | | | |
| | | | | | | | | | | | 0016A | 0016A | P.ABO: | .LONG | 10 | | | | |
| | | | | | | 0000000A | 0016C | | .ADDRESS | P.ABP | | | | | | | | | |
| | | | | | | 00000000 | 00170 | P.ABR: | .ASCII | \HEADER\ | | | | | | | | | |
| | | | 52 | 45 | 44 | 41 | 45 | 48 | | | 00174 | 00174 | P.ABR: | .BLKB | 2 | | | | |
| | | | | | | | | | | | 0017A | 0017A | P.ABQ: | .LONG | 6 | | | | |
| | | | | | | 00000006 | 0017C | | .ADDRESS | P.ABR | | | | | | | | | |
| | | | | | | 00000000 | 00180 | P.ABT: | .ASCII | \TRAILING_SPACES\ | | | | | | | | | |
| 53 | 45 | 43 | 41 | 50 | 53 | 5F | 47 | 4E | 49 | 4C | 49 | 41 | 52 | 54 | 00184 | 00184 | P.ABT: | .BLKB | 1 |
| | | | | | | | | | | | | | | | 00193 | 00193 | P.ABT: | .LONG | 15 |
| | | | | | | 0000000F | 00194 | P.ABS: | .ADDRESS | P.ABT | | | | | | | | | |
| | | | | | | 00000000 | 00198 | | .ASCII | \SPACING\ | | | | | | | | | |
| | | | 47 | 4E | 49 | 43 | 41 | 50 | 53 | | 0019C | 0019C | P.ABV: | .BLKB | 1 | | | | |
| | | | | | | | | | | | 001A3 | 001A3 | P.ABV: | .BLKB | 1 | | | | |

.....

```

00000007 001A4 P.ABU: .LONG 7
00000000' 001A8 .ADDRESS P.ABV
54 43 41 58 45 001AC P.ABX: .ASCII \EXACT\
001B1 .BLKB 3
00000005 001B4 P.ABW: .LONG 5
00000000' 001B8 .ADDRESS P.ABX
59 54 54 45 52 50 001BC P.ABZ: .ASCII \PRETTY\
001C2 .BLKB 2
00000006 001C4 P.ABY: .LONG 6
00000000' 001C8 .ADDRESS P.ABZ
49 49 43 53 41 001CC P.ACB: .ASCII \ASCII\
001D1 .BLKB 3
00000005 001D4 P.ACA: .LONG 5
00000000' 001D8 .ADDRESS P.ACB
4C 41 4D 49 43 45 44 41 58 45 48 001DC P.ACD: .ASCII \HEXADECIMAL\
001E7 .BLKB 1
00000008 001E8 P.ACC: .LONG 11
00000000' 001EC .ADDRESS P.ACD
4C 41 54 43 4F 001F0 P.ACF: .ASCII \OCTAL\
001F5 .BLKB 3
00000005 001F8 P.ACE: .LONG 5
00000000' 001FC .ADDRESS P.ACF
52 45 54 53 41 4D 00200 P.ACH: .ASCII \MASTER\
00206 .BLKB 2
00000006 00208 P.ACG: .LONG 6
00000000' 0020C .ADDRESS P.ACH
4E 4F 49 53 49 56 45 52 00210 P.ACJ: .ASCII \REVISION\
00000008 00218 P.ACI: .LONG 8
00000000' 0021C .ADDRESS P.ACJ

```

.PSECT \$OWNS,NOEXE,2

```

0A 0000 SYSS$OUTPUT:
54 55 50 54 55 4F 24 53 59 53 00001 .BYTE 10
0000B .ASCII \SYSS$OUTPUT\
0000C .BLKB 1
TPALINDESC:
00010 .BLKB 4
TPDATA: .BLKB 4
00014 TPABLOCK:
.BLKB 36
00038 FIND_CONTEXT:
.BLKB 4
00# 0003C FIND_MASFILE:
.BYTE 0[3]
02 0003F .BYTE 2
00040 .BLKB 4

```

```

SD_$LINE= P.AAA
SD_INPUT1= P.AAC
SD_INPUT2= P.AAE
SD_CHANGE_BAR= P.AAG
SD_COMMENT_DELIMITERS=
P.AAI
SD_IGNORE= P.AAK
SD_LINE_WIDTH= P.AAM
SD_MATCH= P.AAO

```

Vertical ellipsis symbols on the right side of the page.

```

SD_MAXIMUM_DIFFERENCES=
SD_MERGED= P.AAQ
SD_MODE= P.AAS
SD_NUMBER= P.AAU
SD_OUTPUT= P.AAW
SD_PARALLEL= P.AAY
SD_SEPARATED= P.ABA
SD_SLP= P.ABE
SD_WIDTH= P.ABG
SD_WINDOW= P.ABI
SD_BLANK_LINES= P.ABK
SD_COMMENTS= P.ABM
SD_FORM_FEEDS= P.ABO
SD_HEADER= P.ABQ
SD_TRAILING_SPACES= P.ABS
SD_SPACING= P.ABU
SD_EXACT= P.ABW
SD_PRETTY= P.ABY
SD_ASCII= P.ACA
SD_HEXADECIMAL= P.ACC
SD_OCTAL= P.ACE
SD_MASTER= P.ACG
SD_REVISION= P.ACI
.EXTRN DIF$GL_TYPCDATA, DIF$GL_COMMDISC
.EXTRN DIF$GL_COMMFLGS
.EXTRN DIF$GL_IGNORE, DIF$GL_CMDESC
.EXTRN DIF$GL_HEADER, DIF$GL_WIDTH
.EXTRN DIF$GL_MATCH, DIF$GL_MAXDIF
.EXTRN DIF$GL_MERGED, DIF$GL_PARALLEL
.EXTRN DIF$GL_WNDSIZ, DIF$GL_FLAGS
.EXTRN DIF$GL_MASFDB, DIF$GL_MASDESC
.EXTRN DIF$GL_MASRAB, DIF$GL_MASNAM
.EXTRN DIF$GL_MASRSS, DIF$GL_REVFDB
.EXTRN DIF$GL_REVDESC, DIF$GL_REVRAB
.EXTRN DIF$GL_REVNAM, DIF$GL_REVRSS
.EXTRN DIF$GL_OUTIFI, DIF$GL_OUTDESC
.EXTRN DIF$GL_OUTRAB, DIF$GL_OUTNAM
.EXTRN DIF$GL_OUTRSS, CLISGET VALUE
.EXTRN CLISPRESENT, LIBSCVT DTB
.EXTRN LIBSFIND FILE, LIB$GET VM
.EXTRN LIB$COPY R DX, LIB$TPARSE
.EXTRN STR$COPY DX, SY$FAO
.EXTRN CLIS PRESENT, CLIS NEGATED
.EXTRN DIF$C_MAXLANGS, DIF$C_BADVALUE
.EXTRN DIF$C_CLOSEIN, DIF$C_CLOSEOUT
.EXTRN DIF$C_CMDLIN, DIF$C_CMDSYNTAX
.EXTRN DIF$C_CONFQUAL, DIF$C_MRGGMTCH
.EXTRN DIF$C_NOLOWVER, DIF$C_NOVALUE
.EXTRN DIF$C_NOWILD, DIF$C_OPENIN
.EXTRN DIF$C_OPENOUT, DIF$C_PARGMTCH
.EXTRN DIF$C_PARSEFAIL, DIF$C_STRCPYERR
.EXTRN DIF$C_SYSFAOERR, DIF$C_TOOMNYDLM
.EXTRN DIF$C_TOOMNYFIL, DIF$C_TWOWIDTHS
.PSECT $CODE$,NOWRT,2

```

| | | | | | | | |
|----|-----------|----|------------|--------------|--------|-----------------------------|--------|
| | | | 007C 00000 | CALL_TPARSE: | | | |
| | | 56 | 00000000' | EF 9E 00002 | .WORD | Save R2,R3,R4,R5,R6 | : 0534 |
| 24 | 00 | 6E | | 00 2C 00009 | MOVAB | TPABLOCK, R6 | : 0556 |
| | | | | 66 0000E | MOVCS | #0, (SP), #0, #36, TPABLOCK | : 0557 |
| | | 66 | | 08 D0 0000F | MOVL | #8, TPABLOCK | : 0558 |
| | 04 | A6 | | 02 D0 00012 | MOVL | #2, TPABLOCK+4 | : 0559 |
| | | 50 | 04 | AC D0 00016 | MOVL | STRING, R0 | : 0560 |
| | 08 | A6 | | 60 3C 0001A | MOVZWL | (R0), TPABLOCK+8 | : 0564 |
| | 0C | A6 | 04 | A0 D0 0001E | MOVL | 4(R0), TPABLOCK+12 | : 0565 |
| | F8 | A6 | | 50 D0 00023 | MOVL | R0, TPALINDESC | : 0567 |
| | | | FC | A6 D4 00027 | CLRL | TPDATA | : 0569 |
| | | 7E | 08 | AC 7D 0002A | MOVQ | STATES, -(SP) | : 0569 |
| | | | | 56 DD 0002E | PUSHL | R6 | : 0569 |
| | 00000000G | 00 | | 03 FB 00030 | CALLS | #3, LIB\$TPARSE | : 0569 |
| | | | | 04 00037 | RET | | |

; Routine Size: 56 bytes, Routine Base: \$CODE\$ + 0000

Print syntax error message from tparse

```

: 271 0570 1 %SBTTL 'Print syntax error message from tparse';
: 272 0571 1 ROUTINE syntaxerr =
: 273 0572 2 BEGIN
: 274 0573 2 |
: 275 0574 2 | This routine is called to print a command syntax error message
: 276 0575 2 | and stops the image.
: 277 0576 2 |
: 278 0577 2 |
: 279 0578 2 SIGNAL_STOP(difs_cmdsyntax,0,
: 280 0579 2 difs_cmdlin,f,..tpalindesc)
: 281 0580 1 END;

```

| | | | | | | |
|--------------|-----------|-------------|-------|-----------------|--|--------|
| | 0000 0000 | SYNTAXERR: | | | | |
| | 00000000' | EF DD 00002 | .WORD | Save nothing | | : 0571 |
| | | 01 DD 00008 | PUSHL | TPALINDESC | | : 0579 |
| | 00000000G | 8F DD 0000A | PUSHL | #1 | | : 0578 |
| | | 7E D4 00010 | CLRL | -(SP) | | : |
| | 00000000G | 8F DD 00012 | PUSHL | #DIFS_CMDSYNTAX | | : |
| 00000000G 00 | | 05 FB 00018 | CALLS | #5, LIB\$STOP | | : |
| | | C4 0001F | RET | | | : 0580 |

; Routine Size: 32 bytes, Routine Base: \$CODE\$ + 0038

DIF_GETCMD
V04=000

Print syntax error message from tparse

L 13
15-Sep-1984 23:39:15
14-Sep-1984 12:19:20

VAX-11 Bliss-32 V4.0-742
[DIF.SRC]DIFGETCMD.B32;1

Page 16
(6)

```

: 283      0581 1 ROUTINE badvaluerr (desc) =
: 284      0582 2 BEGIN
: 285      0583 3
: 286      0584 4 ! Print an error message about a bad qualifier value and stop the image
: 287      0585 5
: 288      0586 6 SIGNAL_STOP(dif$_badvalue,1,..desc)
: 289      0587 7 END;

```

```

                                0000 00000 BADVALUERR:
                                04 AC DD 00002      .WORD Save nothing
                                01 DD 00005      PUSHL DESC
                                8F DD 00007      PUSHL #1
00000000G 00 00000000G 03 FB 0000D      PUSHL #DIF$_BADVALUE
                                04 00014      CALLS #3, LIB$STOP
                                RET

```

```

: 0581
: 0586
: 0587

```

; Routine Size: 21 bytes. Routine Base: \$CODE\$ + 0058

Get change bar qualifier

```

291 0588 1 %SBTTL 'Get change bar qualifier';
292 0589 1 ROUTINE getchbarqual (fdb) =
293 0590 2 BEGIN
294 0591 2
295 0592 2 Get the /CHANGE BAR qualifier if specified. This is a file-specific
296 0593 2 qualifier. A cli$get_value for the appropriate input file specification
297 0594 2 should be done immediately prior to calling this routine
298 0595 2
299 0596 2 Inputs:
300 0597 2
301 0598 2 fdb address of selected input file fdb
302 0599 2
303 0600 2 MAP
304 0601 2 fdb : REF BBLOCK;
305 0602 2
306 0603 2 LOCAL
307 0604 2 numberkey,
308 0605 2 desc : desc_decl;
309 0606 2
310 0607 2 BIND
311 0608 2 cchar = desc[dsc$a_pointer] : REF VECTOR[,BYTE];
312 0609 2
313 0610 2 fdb[fdb$b_cbarchr] = %ASCII '!'; !Set default character
314 0611 2 numberkey = false;
315 0612 2 IF (fdb[fdb$v_changebar] = cli$present(sd_change_bar))
316 0613 2 THEN BEGIN
317 0614 2 CH$FILL(0,dsc$c_s_bln,desc); !Make a dynamic
318 0615 2 desc[dsc$b_class] = dsc$k_class_d; ! string descriptor
319 0616 2 WHILE cli$get_value(sd_change_bar,desc) !Get value
320 0617 2 DO BEGIN
321 0618 2 IF call tparse(desc,linenum_states,linenum_keys)
322 0619 2 THEN BEGIN
323 0620 2 fdb[fdb$v_linenum] = .tpdata;
324 0621 2 numberkey = true;
325 0622 2 END
326 0623 2 ELSE IF .desc[dsc$w_length] NEQ 0
327 0624 2 THEN fdb[fdb$b_cbarchr] = .cchar[0];
328 0625 2 END;
329 0626 2 IF NOT .numberkey
330 0627 2 THEN fdb[fdb$v_linenum] = .dif$gl_flags[dif$v_linenum];
331 0628 2 END;
332 0629 2
333 0630 2 RETURN true
334 0631 1 END;

```

| | | | | | | | |
|----|----|-----------|-----------|---------------|-------|---------------------------|--------|
| | | | 01FC 0000 | GETCHBARQUAL: | | | |
| | | | | | .WORD | Save R2,R3,R4,R5,R6,R7,R8 | : 0589 |
| | 58 | 00000000' | EF | 9E 00002 | MOVAB | SD_CHANGE_BAR, R8 | : |
| | 5E | | 08 | C2 00009 | SUBL2 | #8, SP | : |
| | 56 | 04 | AC | D0 0000C | MOVL | FDB, R6 | : 0610 |
| 26 | A6 | | 21 | 90 00010 | MOVB | #33, 38(R6) | : |
| | | | 57 | D4 00014 | CLRL | NUMBERKEY | : 0611 |
| | | | 58 | DD 00016 | PUSHL | R3 | : 0612 |

| | | | | | | | | | | |
|----|----|-----------|-----------|----|----|-------|-------|--------|-----------------------------|------|
| 24 | A6 | 01 | 00000000G | 00 | 01 | FB | 00018 | CALLS | #1, CLISPRESNT | |
| | | | | 00 | 50 | F0 | 0001F | INSV | RO, #0, #1, 36(R6) | |
| | 08 | 00 | | 5B | 50 | E9 | 00025 | BLBC | RO, 4\$ | |
| | | | | 6E | 00 | 2C | 00028 | MOVCS | #0, (SP), #0, #8, DESC | 0614 |
| | | | 03 | AE | 6E | | 0002D | | | |
| | | | | | 02 | 90 | 0002E | MOVB | #2, DESC+3 | 0615 |
| | | | | | 8F | E3 | 00032 | PUSHR | #M<R8, SP> | 0616 |
| | | | 00000000G | 00 | 02 | FB | 00036 | CALLS | #2, CLISGET_VALUE | |
| | | | | 31 | 50 | E9 | 0003D | BLBC | RO, 3\$ | |
| | | | | | EF | 9F | 00040 | PUSHAB | LINENUM_KEYS | 0618 |
| | | | | | EF | 9F | 00046 | PUSHAB | LINENUM_STATES | |
| | | | | | AE | 9F | 0004C | PUSHAB | DESC | |
| | | | FF3F | CF | 03 | FB | 0004F | CALLS | #3, CALL_TPARSE | |
| | | | | 0F | 50 | E9 | 00054 | BLBC | RO, 2\$ | |
| 24 | A6 | 01 | | 01 | EF | F0 | 00057 | INSV | TPDATA, #1, #1, 36(R6) | 0620 |
| | | | | 57 | 01 | D0 | 00061 | MOVL | #1, NUMBERKEY | 0621 |
| | | | | | CC | 11 | 00064 | BRB | 1\$ | 0618 |
| | | | | | 6E | B5 | 00066 | TSTW | DESC | 0623 |
| | | | | | C8 | 13 | 00068 | BEQL | 1\$ | |
| | | | 26 | A6 | 04 | BE | 0006A | MOVB | @CCHAR, 38(R6) | 0624 |
| | | | | | C1 | 11 | 0006F | BRB | 1\$ | 0616 |
| | | | | 0F | 57 | E8 | 00071 | BLBS | NUMBERKEY, 4\$ | 0625 |
| 24 | 50 | 00000000G | 00 | 01 | 04 | EF | 00074 | EXTZV | #4, #1, DIF\$GL_FLAGS+1, RO | 0627 |
| | A6 | | 01 | 01 | 50 | F0 | 0007D | INSV | RO, #1, #1, 36(R6) | |
| | | | | 50 | 01 | D0 | 00083 | MOVL | #1, RO | 0630 |
| | | | | | 04 | 00086 | RET | | | 0631 |

; Routine Size: 135 bytes, Routine Base: \$CODE\$ + 006D

DIF\$GETCMD Process command line

```
336 0632 1 %SBTTL 'DIF$GETCMD Process command line';
337 0633 1 GLOBAL ROUTINE dif$getcmd =
338 0634 2 BEGIN
339 0635 2
340 0636 2   Get the command parameters
341 0637 2
342 0638 2   Inputs:
343 0639 2
344 0640 2     none
345 0641 2
346 0642 2   Implicit inputs:
347 0643 2
348 0644 2     command line entered, image invoked by DCL
349 0645 2
350 0646 2 LOCAL
351 0647 2   status,
352 0648 2   desc : desc_decl;
353 0649 2
354 0650 2 CH$FILL(0,dsc$c_s_bln,desc);           ! Make desc a dynamic string descriptor
355 0651 2 desc[dsc$b_class] = dsc$k_class_d;
356 0652 2
357 0653 2
358 0654 2   Get the /NUMBER and /SLP qualifiers
359 0655 2
360 0656 2 dif$gl_flags[dif$v_linenum] = cli$present(sd_number);
361 0657 2 dif$gl_flags[dif$v_slp] = cli$present(sd_slp);
362 0658 2
363 0659 2
364 0660 2   Get the /SEPARATED qualifier
365 0661 2
366 0662 2 dif$gl_flags [dif$v_separated] = cli$present(sd_separated);
367 0663 2 dif$gl_masfdb [fdb$v_separated] = cli$present(sd_master);
368 0664 2 dif$gl_revfdb [fdb$v_separated] = cli$present(sd_revision);
369 0665 2
370 0666 2
371 0667 2   Get the /LINE_WIDTH or /WIDTH qualifier
372 0668 2
373 0669 2 IF cli$get_value(sd_line_width,desc)
374 0670 2 THEN BEGIN
375 0671 2   status = lib$cvt_dtb(.desc[dsc$w_length],
376 0672 2     .desc[dsc$a_pointer],dif$gl_width);
377 0673 2   IF NOT .status
378 0674 2     THEN badvaluerr(desc);
379 0675 2   dif$gl_flags[dif$v_width] = true;
380 0676 2   END;
381 0677 2
382 0678 2 IF cli$get_value(sd_width,desc)
383 0679 2 THEN BEGIN
384 0680 2   status = lib$cvt_dtb(.desc[dsc$w_length],
385 0681 2     .desc[dsc$a_pointer],dif$gl_width);
386 0682 2   IF NOT .status
387 0683 2     THEN badvaluerr(desc);
388 0684 2   dif$gl_flags[dif$v_width] = true;
389 0685 2   END;
390 0686 2
391 0687 2 IF cli$present(sd_line_width) and cli$present(sd_width)
392 0688 2 THEN SIGNAL (dif$twowidths);
```

```

393 0689 2
394 0690 2
395 0691 2 Get the /MATCH qualifier
396 0692 2
397 0693 2 IF (dif$gl_flags[dif$v_match] = cli$get_value(sd_match,desc))
398 0694 2 THEN BEGIN
399 0695 2     status = lib$cvdtb(.desc[dsc$w_length],
400 0696 2         .desc[dsc$a_pointer],dif$gl_match);
401 0697 2     IF NOT .status
402 0698 2         THEN badvaluerr(desc);
403 0699 2     END;
404 0700 2
405 0701 2
406 0702 2 Get the /MAXIMUM_DIFFERENCES qualifier
407 0703 2
408 0704 2 IF (dif$gl_flags[dif$v_maxdif] = cli$get_value(sd_maximum_differences,desc))
409 0705 2 THEN BEGIN
410 0706 2     status = lib$cvdtb(.desc[dsc$w_length],
411 0707 2         .desc[dsc$a_pointer],dif$gl_maxdif);
412 0708 2     IF NOT .status
413 0709 2         THEN badvaluerr(desc);
414 0710 2     END;
415 0711 2
416 0712 2
417 0713 2 Get the /MERGED qualifier
418 0714 2
419 0715 2 IF cli$present(sd_merged)
420 0716 2 THEN BEGIN
421 0717 2     dif$gl_flags[dif$v_merged] = true;
422 0718 2     IF cli$get_value(sd_merged,desc)
423 0719 2         AND NOT (status = lib$cvdtb(.desc[dsc$w_length],
424 0720 2             .desc[dsc$a_pointer],
425 0721 2             dif$gl_merged))
426 0722 2         THEN badvaluerr(desc);
427 0723 2     END;
428 0724 2
429 0725 2
430 0726 2 Get the /PARALLEL qualifier
431 0727 2
432 0728 2 IF cli$present(sd_parallel)
433 0729 2 THEN BEGIN
434 0730 2     dif$gl_flags[dif$v_parallel] = true;
435 0731 2     IF cli$get_value(sd_parallel,desc)
436 0732 2         AND NOT (status = lib$cvdtb(.desc[dsc$w_length],
437 0733 2             .desc[dsc$a_pointer],
438 0734 2             dif$gl_parallel))
439 0735 2         THEN badvaluerr(desc);
440 0736 2     END;
441 0737 2
442 0738 2
443 0739 2 Get the /WINDOW qualifier
444 0740 2
445 0741 2 IF (dif$gl_flags[dif$v_window] = cli$get_value(sd_window,desc))
446 0742 2 THEN BEGIN
447 0743 2     status = lib$cvdtb(.desc[dsc$w_length],
448 0744 2         .desc[dsc$a_pointer],dif$gl_wndwsiz);
449 0745 2     IF ((NOT .status) OR (.dif$gl_wndwsiz LEQ 0))

```

```

: 450      0746      3      THEN badvaluerr(desc);
: 451      0747      2      END;
: 452      0748      2
: 453      0749      2
: 454      0750      2      Get the /COMMENT_DELIMITERS qualifier
: 455      0751      2
: 456      0752      3      IF (dif$gl_flags[dif$v_comdel] = cli$present(sd_comment_delimiters))
: 457      0753      3      THEN BEGIN
: 458      0754      3          LOCAL
: 459      0755      3              delimbuf = VECTOR[dif$c_maxcdlm, BYTE],
: 460      0756      3              delimindex;
: 461      0757      3
: 462      0758      3              delimindex = 0;
: 463      0759      3              WHILE cli$get_value(sd_comment_delimiters, desc)
: 464      0760      4              DO BEGIN
: 465      0761      4                  IF .desc [dsc$w_length] GTRU 1
: 466      0762      6                      THEN (IF NOT (status = call_tparse(desc, comment_delim_states, delim_keys))
: 467      0763      5                          THEN syntaxerr())
: 468      0764      4                      ELSE CHSMOVE (1, .desc [dsc$a_pointer], tpdata);
: 469      0765      4                  IF .delimindex EQL dif$c_maxcdlm
: 470      0766      4                      THEN SIGNAL_STOP(dif$t_toomnydlm, 1, dif$c_maxcdlm);
: 471      0767      4                  delimbuf[.delimindex] = .tpdata;
: 472      0768      4                  delimindex = .delimindex + 1;
: 473      0769      3              END;
: 474      0770      3              IF .delimindex NEQ 0
: 475      0771      4                  THEN IF NOT (status = lib$scopy_r_dx(delimindex, delimbuf, dif$gl_commdesc))
: 476      0772      3                      THEN SIGNAL_STOP(dif$_strcpyerr, 0, .status);
: 477      0773      2              END;
: 478      0774      2
: 479      0775      2      Get the /IGNORE qualifier
: 480      0776      2
: 481      0777      2
: 482      0778      2      dif$gl_flags[dif$v_ignore] = cli$present(sd_ignore);
: 483      0779      2      dif$gl_ignore [ign$v_blklin] = cli$present(sd_blank_lines);
: 484      0780      2      dif$gl_ignore [ign$v_comments] = cli$present(sd_comments);
: 485      0781      2      dif$gl_ignore [ign$v_exact] = cli$present(sd_exact);
: 486      0782      2      dif$gl_ignore [ign$v_formfeed] = cli$present(sd_form_feeds);
: 487      0783      3      IF (dif$gl_ignore [ign$v_header] = cli$present(sd_header))
: 488      0784      3          THEN BEGIN
: 489      0785      3              IF cli$get_value(sd_header, desc)
: 490      0786      4                  AND NOT (status = lib$cvt_dtb(.desc[dsc$w_length],
: 491      0787      4                      .desc[dsc$a_pointer],
: 492      0788      4                      dif$gl_header))
: 493      0789      3                      THEN badvaluerr(desc);
: 494      0790      3              dif$gl_masfdb [fdb$l_headcnt] = .dif$gl_header;
: 495      0791      3              dif$gl_revfdb [fdb$l_headcnt] = .dif$gl_header;
: 496      0792      2              END;
: 497      0793      2      dif$gl_ignore [ign$v_pretty] = cli$present(sd_pretty);
: 498      0794      2      dif$gl_ignore [ign$v_traiblnk] = cli$present(sd_trailing_spaces);
: 499      0795      2      dif$gl_ignore [ign$v_spacing] = cli$present(sd_spacing);
: 500      0796      2
: 501      0797      2
: 502      0798      2      IF .dif$gl_flags [dif$v_comdel]
: 503      0799      3          THEN BEGIN
: 504      0800      3              dif$gl_flags [dif$v_ignore] = true;
: 505      0801      3              dif$gl_ignore [ign$v_comments] = true;
: 506      0802      2          END;
```

DIF\$GETCMD Process command line

```

507 0803 2
508 0804 2
509 0805 2 : Get the /MODE qualifier
510 0806 2
511 0807 2 dif$gl_flags[dif$v_ascii] = cli$present(sd_ascii);
512 0808 2 dif$gl_flags[dif$v_hex] = cli$present(sd_hexadecimal);
513 0809 2 dif$gl_flags[dif$v_octal] = cli$present(sd_octal);
514 0810 2
515 0811 2
516 0812 2 : Get the /OUTPUT qualifier
517 0813 2
518 0814 2 IF (dif$gl_flags[dif$v_output] = cli$present(sd_output))
519 0815 2 THEN cli$get_value(sd_output,dif$gl_outdesc)
520 0816 2 ELSE IF NOT .dif$gl_flags[dif$v_slp]
521 0817 2 THEN lib$scopy_r_dx(%REF-.sys$output [0]), sys$output [1],
522 0818 2 dif$gl_outdesc);
523 0819 2
524 0820 2 : Get the input files
525 0821 2
526 0822 2
527 0823 2 cli$get_value(sd_input1,find_masfile);
528 0824 2 getchbarqual(dif$gl_masfdb); !Get /CHANGE_BAR for master file
529 0825 2
530 0826 2
531 0827 2 : Get value in case two params specified on first input
532 0828 2
533 0829 2 IF (status = cli$get_value(sd_input1,desc)) !Get 2nd file on first parameter
534 0830 2 THEN getchbarqual(dif$gl_revfdb);
535 0831 2 IF cli$get_value(sd_input2,dif$gl_revdesc) !If 2nd parameter specified
536 0832 2 THEN BEGIN
537 0833 2 IF .status ! and two files given on first param
538 0834 2 THEN SIGNAL_STOP(dif$_toomnyfil);
539 0835 2 END
540 0836 2 ELSE IF .status !2nd param not specified
541 0837 2 THEN CH$MOVE(dsc$_s_bln,desc,dif$gl_revdesc); ! but 2 files on first param were, so use it
542 0838 2 getchbarqual(dif$gl_revfdb); !Get /CHANGE_BAR for revision file
543 0839 2
544 0840 2 : Get the command line
545 0841 2
546 0842 2
547 0843 2 cli$get_value(sd_$line,dif$gl_cmdesc);
548 0844 2
549 0845 2 : Check for conflicting qualifiers
550 0846 2
551 0847 2
552 0848 2 IF .dif$gl_flags[dif$v_parallel] !If /parallel specified
553 0849 2 THEN dif$gl_flags[dif$v_ascii] = true; !Force ASCII mode
554 0850 2
555 0851 2 IF .dif$gl_flags[dif$v_slp] !If /slp specified
556 0852 2 THEN BEGIN
557 0853 2 dif$gl_ignore[ign$v_exact] = true; !Use unedited text
558 0854 2 dif$gl_ignore[ign$v_pretty] = false; !Don't expand tabs, etc.
559 0855 2 dif$gl_flags[dif$v_inenum] = false; !Don't output numbers
560 0856 2 END
561 0857 2 ELSE IF NOT .dif$gl_flags[dif$v_separated] !Not /slp, if not /separated
562 0858 2 AND NOT .dif$gl_masfdb[fdb$v_changebar] ! or /changebar
563 0859 2 AND NOT .dif$gl_revfdb[fdb$v_changebar] ! on either file

```



```

: 564 0860 2 AND NOT .dif$gl_flags[dif$v_parallel] : or /parallel
: 565 0861 2 THEN dif$gl_flags[dif$v_merged] = true; : then set /merged
: 566 0862 2
: 567 0863 2 : The values of /MERGE and /PARALLEL must be less than /MATCH
: 568 0864 2
: 569 0865 2 IF .dif$gl_merged GTR .dif$gl_match
: 570 0866 2 THEN BEGIN
: 571 0867 2 SIGNAL(dif$mrggtmtch,2,.dif$gl_merged,.dif$gl_match);
: 572 0868 2 dif$gl_merged = .dif$gl_match;
: 573 0869 2 END;
: 574 0870 2
: 575 0871 2 IF .dif$gl_parallel GTR .dif$gl_match
: 576 0872 2 THEN BEGIN
: 577 0873 2 SIGNAL(dif$pargtmtch,2,.dif$gl_parallel,.dif$gl_match);
: 578 0874 2 dif$gl_parallel = .dif$gl_match;
: 579 0875 2 END;
: 580 0876 2
: 581 0877 2 RETURN true
: 582 0878 1 END;

```

| | | | | | | | |
|-----------|----|----|-----------|--------------------------|--------|---|------|
| | | | | OFFC 00000 | .ENTRY | DIF\$GETCMD, Save R2,R3,R4,R5,R6,R7,R8,R9,- | 0633 |
| | | | | | | R10,R11 | |
| | | | | 5B 00000000G 00 9E 00002 | MOVAB | DIF\$GL IGNORE, R11 | |
| | | | | 5A FF57 CF 9E 00009 | MOVAB | BADVALDERR, R10 | |
| | | | | 59 00000000G 00 9E 0000E | MOVAB | CLISGET VALUE, R9 | |
| | | | | 58 00000000G 00 9E 00015 | MOVAB | CLISPRESENT, R8 | |
| | | | | 57 00000000G 00 9E 0001C | MOVAB | DIF\$GL FLAGS, R7 | |
| | | | | 56 00000000' EF 9E 00023 | MOVAB | SD LINE_WIDTH, R6 | |
| | | | | 5E 30 C2 0002A | SUBL2 | #48, SP | |
| 08 | | 00 | | 6E 00 2C 0002D | MOVCS | #0, (SP), #0, #8, DESC | 0650 |
| | | | 28 | AE 00032 | | | |
| | | | 2B | AE 02 90 00034 | MOVAB | #2, DESC+3 | 0651 |
| | | | 58 | A6 9F 00038 | PUSHAB | SD_NUMBER | 0656 |
| | | | 68 | 01 FB 0003B | CALLS | #1, CLISPRESENT | |
| 01 | A7 | 01 | 04 | 50 F0 0003E | INSV | R0, #4, #1, DIF\$GL_FLAGS+1 | |
| | | | 0098 | C6 9F 00044 | PUSHAB | SD_SLP | 0657 |
| | | | 68 | 01 FB 00048 | CALLS | #1, CLISPRESENT | |
| 01 | A7 | 01 | 00 | 50 F0 0004B | INSV | R0, #0, #1, DIF\$GL_FLAGS+1 | |
| | | | 008C | C6 9F 00051 | PUSHAB | SD_SEPARATED | 0662 |
| | | | 68 | 01 FB 00055 | CALLS | #1, CLISPRESENT | |
| | | | 07 | 50 F0 00058 | INSV | R0, #7, #1, DIF\$GL_FLAGS | |
| | | | 018C | C6 9F 0005D | PUSHAB | SD_MASTER | 0663 |
| | | | 68 | 01 FB 00061 | CALLS | #1, CLISPRESENT | |
| 00000000G | 00 | 01 | 02 | 50 F0 00064 | INSV | R0, #2, #1, DIF\$GL_MASFDB+36 | |
| | | | 019C | C6 9F 0006D | PUSHAB | SD_REVISION | 0664 |
| | | | 68 | 01 FB 00071 | CALLS | #1, CLISPRESENT | |
| 00000000G | 00 | 01 | 02 | 50 F0 00074 | INSV | R0, #2, #1, DIF\$GL_REVFDB+36 | |
| | | | 28 | AE 9F 0007D | PUSHAB | DESC | 0669 |
| | | | | 56 DD 00080 | PUSHL | R6 | |
| | | | 69 | 02 FB 00082 | CALLS | #2, CLISGET_VALUE | |
| | | | 24 | 50 E9 00085 | BLBC | R0, 28 | |
| | | | 00000000G | 00 9F 00088 | PUSHAB | DIF\$GL_WIDTH | 0671 |
| | | | 30 | AE DD 0008E | PUSHL | DESC+4 | 0672 |

| | | | | | | | | | |
|--|-----------|----|-----------|----|----|-------|-------------|----------------------------|------|
| | | 7E | 30 | AE | 3C | 00091 | MOVZWL | DESC, -(SP) | 0671 |
| | 00000000G | 00 | | 03 | FB | 00095 | CALLS | #3, LIBSCVT_DTB | |
| | | 52 | | 50 | DO | 0009C | MOVL | R0, STATUS | |
| | | 06 | | 52 | EB | 0009F | BLBS | STATUS, 1\$ | 0673 |
| | | | 28 | AE | 9F | 000A2 | PUSHAB | DESC | 0674 |
| | | 6A | | 01 | FB | 000A5 | CALLS | #1, BADVALUERR | |
| | 01 | A7 | | 02 | 88 | 000A8 | 1\$: BISB2 | #2, DIFSGL_FLAGS+1 | 0675 |
| | | | 28 | AE | 9F | 000AC | 2\$: PUSHAB | DESC | 0678 |
| | | | 00AB | C6 | 9F | 000AF | PUSHAB | SD_WIDTH | |
| | | 69 | | 02 | FB | 000B3 | CALLS | #2, CLISGET_VALUE | |
| | | 24 | | 50 | E9 | 000B6 | BLBC | R0, 4\$ | |
| | | | 00000000G | 00 | 9F | 000B9 | PUSHAB | DIFSGL_WIDTH | 0680 |
| | | | 30 | AE | DD | 000BF | PUSHL | DESC+4 | 0681 |
| | 00000000G | 7E | 30 | AE | 3C | 000C2 | MOVZWL | DESC, -(SP) | 0680 |
| | | 00 | | 03 | FB | 000C6 | CALLS | #3, LIBSCVT_DTB | |
| | | 52 | | 50 | DO | 000CD | MOVL | R0, STATUS | |
| | | 06 | | 52 | EB | 000D0 | BLBS | STATUS, 3\$ | 0682 |
| | | | 28 | AE | 9F | 000D3 | PUSHAB | DESC | 0683 |
| | | 6A | | 01 | FB | 000D6 | CALLS | #1, BADVALUERR | |
| | 01 | A7 | | 02 | 88 | 000D9 | 3\$: BISB2 | #2, DIFSGL_FLAGS+1 | 0684 |
| | | | | 56 | DD | 000DD | 4\$: PUSHL | R6 | 0687 |
| | | 68 | | 01 | FB | 000DF | CALLS | #1, CLISPRESENT | |
| | | 17 | | 50 | E9 | 000E2 | BLBC | R0, 5\$ | |
| | | | 00AB | C6 | 9F | 000E5 | PUSHAB | SD_WIDTH | |
| | | 68 | | 01 | FB | 000E9 | CALLS | #1, CLISPRESENT | |
| | | 0D | | 50 | E9 | 000EC | BLBC | R0, 5\$ | |
| | | | 00000000G | 8F | DD | 000EF | PUSHL | #DIFS_TWOWIDTHS | 0688 |
| | 00000000G | 00 | | 01 | FB | 000F5 | CALLS | #1, LIBSSIGNAL | |
| | | | 28 | AE | 9F | 000FC | 5\$: PUSHAB | DESC | 0693 |
| | | | 10 | A6 | 9F | 000FF | PUSHAB | SD_MATCH | |
| | | 69 | | 02 | FB | 00102 | CALLS | #2, CLISGET_VALUE | |
| | 01 | A7 | 01 | 50 | F0 | 00105 | INSV | R0, #2, #1, DIFSGL_FLAGS+1 | |
| | | 20 | | 50 | E9 | 0010B | BLBC | R0, 6\$ | |
| | | | 00000000G | 00 | 9F | 0010E | PUSHAB | DIFSGL_MATCH | 0695 |
| | | | 30 | AE | DD | 00114 | PUSHL | DESC+4 | 0696 |
| | 00000000G | 7E | 30 | AE | 3C | 00117 | MOVZWL | DESC, -(SP) | 0695 |
| | | 00 | | 03 | FB | 0011B | CALLS | #3, LIBSCVT_DTB | |
| | | 52 | | 50 | DO | 00122 | MOVL | R0, STATUS | |
| | | 06 | | 52 | EB | 00125 | BLBS | STATUS, 6\$ | 0697 |
| | | | 28 | AE | 9F | 00128 | PUSHAB | DESC | 0698 |
| | | 6A | | 01 | FB | 0012B | CALLS | #1, BADVALUERR | |
| | | | 28 | AE | 9F | 0012E | 6\$: PUSHAB | DESC | 0704 |
| | | | 2C | A6 | 9F | 00131 | PUSHAB | SD_MAXIMUM_DIFFERENCES | |
| | | 69 | | 02 | FB | 00134 | CALLS | #2, CLISGET_VALUE | |
| | 01 | A7 | 01 | 50 | F0 | 00137 | INSV | R0, #6, #1, DIFSGL_FLAGS+1 | |
| | | 20 | | 50 | E9 | 0013D | BLBC | R0, 7\$ | |
| | | | 00000000G | 00 | 9F | 00140 | PUSHAB | DIFSGL_MAXDIF | 0706 |
| | | | 30 | AE | DD | 00146 | PUSHL | DESC+4 | 0707 |
| | 00000000G | 7E | 30 | AE | 3C | 00149 | MOVZWL | DESC, -(SP) | 0706 |
| | | 00 | | 03 | FB | 0014D | CALLS | #3, LIBSCVT_DTB | |
| | | 52 | | 50 | DO | 00154 | MOVL | R0, STATUS | |
| | | 06 | | 52 | EB | 00157 | BLBS | STATUS, 7\$ | 0708 |
| | | | 28 | AE | 9F | 0015A | PUSHAB | DESC | 0709 |
| | | 6A | | 01 | FB | 0015D | CALLS | #1, BADVALUERR | |
| | | | 3C | A6 | 9F | 00160 | 7\$: PUSHAB | SD_MERGED | 0715 |
| | | 68 | | 01 | FB | 00163 | CALLS | #1, CLISPRESENT | |
| | | 2F | | 50 | E9 | 00166 | BLBC | R0, 8\$ | |

| | | | | | | | | | | |
|----|-----------|----|-----------|-----------|----|-------|--------|-----------------------------|-----------------------|------|
| | | 67 | | 20 | 88 | 00169 | BISB2 | #32, DIF\$GL_FLAGS | 0717 | |
| | | | 28 | AE | 9F | 0016C | PUSHAB | DESC | 0718 | |
| | | | 3C | A6 | 9F | 0016F | PUSHAB | SD_MERGED | | |
| | | 69 | | 02 | FB | 00172 | CALLS | #2, CLISGET_VALUE | | |
| | | 20 | | 50 | E9 | 00175 | BLBC | R0, 8\$ | | |
| | | | 00000000G | 00 | 9F | 00178 | PUSHAB | DIF\$GL_MERGED | 0719 | |
| | | | | 30 | AE | DD | 0017E | PUSHL | DESC+4- | 0720 |
| | | 7E | | 30 | AE | 3C | 00181 | MOVZWL | DESC, -(SP) | 0719 |
| | 00000000G | 00 | | 03 | FB | 00185 | CALLS | #3, LIB\$CVT_DTB | | |
| | | 52 | | 50 | DO | 0018C | MOVL | R0, STATUS | | |
| | | 06 | | 52 | E8 | 0018F | BLBS | STATUS, 8\$ | | |
| | | | | 28 | AE | 9F | 00192 | PUSHAB | DESC | 0722 |
| | | 6A | | 01 | FB | 00195 | CALLS | #1, BADVALUERR | | |
| | | | | 78 | A6 | 9F | 00198 | PUSHAB | SD_PARALLEL | 0728 |
| | | 68 | | 01 | FB | 0019B | CALLS | #1, CLISPRESENT | | |
| | | 30 | | 50 | E9 | 0019E | BLBC | R0, 9\$ | | |
| | | 67 | | 40 | 8F | 88 | 001A1 | BISB2 | #64, DIF\$GL_FLAGS | 0730 |
| | | | | 28 | AE | 9F | 001A5 | PUSHAB | DESC | 0731 |
| | | | | 78 | A6 | 9F | 001A8 | PUSHAB | SD_PARALLEL | |
| | | 69 | | 02 | FB | 001AB | CALLS | #2, CLISGET_VALUE | | |
| | | 20 | | 50 | E9 | 001AE | BLBC | R0, 9\$ | | |
| | | | | 00000000G | 00 | 9F | 001B1 | PUSHAB | DIF\$GL_PARALLEL | 0732 |
| | | | | 30 | AE | DD | 001B7 | PUSHL | DESC+4- | 0733 |
| | | 7E | | 30 | AE | 3C | 001BA | MOVZWL | DESC, -(SP) | 0732 |
| | 00000000G | 00 | | 03 | FB | 001BE | CALLS | #3, LIB\$CVT_DTB | | |
| | | 52 | | 50 | DO | 001C5 | MOVL | R0, STATUS | | |
| | | 06 | | 52 | E8 | 001C8 | BLBS | STATUS, 9\$ | | |
| | | | | 28 | AE | 9F | 001CB | PUSHAB | DESC | 0735 |
| | | 6A | | 01 | FB | 001CE | CALLS | #1, BADVALUERR | | |
| | | | | 28 | AE | 9F | 001D1 | PUSHAB | DESC | 0741 |
| | | | | 00B8 | C6 | 9F | 001D4 | PUSHAB | SD_WINDOW | |
| | | 69 | | 02 | FB | 001D8 | CALLS | #2, CLISGET_VALUE | | |
| 01 | A7 | 01 | | 50 | F0 | 001DB | INSV | R0, #3, #1, DIF\$GL_FLAGS+1 | | |
| | | 03 | | 50 | E9 | 001E1 | BLBC | R0, 11\$ | | |
| | | 28 | | 00000000G | 00 | 9F | 001E4 | PUSHAB | DIF\$GL_WNDWSIZ | 0743 |
| | | | | 30 | AE | DD | 001EA | PUSHL | DESC+4- | 0744 |
| | | 7E | | 30 | AE | 3C | 001ED | MOVZWL | DESC, -(SP) | 0743 |
| | 00000000G | 00 | | 03 | FB | 001F1 | CALLS | #3, LIB\$CVT_DTB | | |
| | | 52 | | 50 | DO | 001F8 | MOVL | R0, STATUS | | |
| | | 08 | | 52 | E9 | 001FB | BLBC | STATUS, 10\$ | 0745 | |
| | | | | 00000000G | 00 | D5 | 001FE | TSTL | DIF\$GL_WNDWSIZ | |
| | | | | 06 | 14 | 00204 | BGTR | 11\$ | | |
| | | | | 28 | AE | 9F | 00206 | PUSHAB | DESC | 0746 |
| | | 6A | | 01 | FB | 00209 | CALLS | #1, BADVALUERR | | |
| | | | | DC | A6 | 9F | 0020C | PUSHAB | SD_COMMENT_DELIMITERS | 0752 |
| | | 68 | | 01 | FB | 0020F | CALLS | #1, CLISPRESENT | | |
| | | 03 | | 50 | F0 | 00212 | INSV | R0, #3, #1, DIF\$GL_FLAGS | | |
| 67 | | 01 | | 50 | E8 | 00217 | BLBS | R0, 12\$ | | |
| | | | | 0094 | 31 | 0021A | BRW | 18\$ | | |
| | | | | 04 | AE | D4 | 0021D | CLRL | DELIMINDEX | 0758 |
| | | | | 28 | AE | 9F | 00220 | PUSHAB | DESC | 0759 |
| | | | | DC | A6 | 9F | 00223 | PUSHAB | SD_COMMENT_DELIMITERS | |
| | | 69 | | 02 | FB | 00226 | CALLS | #2, CLISGET_VALUE | | |
| | | 56 | | 50 | E9 | 00229 | BLBC | R0, 17\$ | | |
| | | 01 | | 28 | AE | B1 | 0022C | CMPW | DESC, #1 | 0761 |
| | | | | | 1F | 1B | 00230 | BLEQU | 14\$ | |
| | | | | 00000000' | EF | 9F | 00232 | PUSHAB | DELIM_KEYS | 0762 |

| | | | | | | | | | | |
|----|----|-----------|-----------|----|----|-------|--------|-----------------------------|-----------------------|------|
| | | 52 | | 50 | DO | 0031C | MOVL | R0, STATUS | | |
| | | 06 | | 52 | EB | 0031F | BLBS | STATUS, 19\$ | | |
| | | | 28 | AE | 9F | 00322 | PUSHAB | DESC | | 0789 |
| | | 6A | | 01 | FB | 00325 | CALLS | #1, BADVALUERR | | |
| | | 50 | 00000000G | 00 | DO | 00328 | 19\$: | MOVL | DIF\$GL_HEADER, R0 | 0790 |
| | | 00 | 00000000G | 50 | DO | 0032F | MOVL | R0, DIF\$GL_MASFDB+40 | | |
| | | 00 | | 50 | DO | 00336 | MOVL | R0, DIF\$GL_REVFDB+40 | | 0791 |
| | | | 0148 | C6 | 9F | 0033D | 20\$: | PUSHAB | SD_PRETTY | 0793 |
| 6B | | 68 | | 01 | FB | 00341 | CALLS | #1, CLISPRESENT | | |
| | 01 | 07 | | 50 | FO | 00344 | INSV | R0, #7, #1, DIF\$GL_IGNORE | | |
| | | | 0118 | C6 | 9F | 00349 | PUSHAB | SD_TRAILING_SPACES | | 0794 |
| 6B | | 68 | | 01 | FB | 0034D | CALLS | #1, CLISPRESENT | | |
| | 01 | 03 | | 50 | FO | 00350 | INSV | R0, #3, #1, DIF\$GL_IGNORE | | |
| | | | 0128 | C6 | 9F | 00355 | PUSHAB | SD_SPACING | | 0795 |
| 6B | | 68 | | 01 | FB | 00359 | CALLS | #1, CLISPRESENT | | |
| | 01 | 04 | | 50 | FO | 0035C | INSV | R0, #4, #1, DIF\$GL_IGNORE | | |
| | 06 | 67 | | 03 | E1 | 00361 | BBC | #3, DIF\$GL_FLAGS, 21\$ | | 0798 |
| | | 67 | | 10 | 88 | 00365 | BISB2 | #16, DIF\$GL_FLAGS | | 0800 |
| | | 6B | | 02 | 88 | 00368 | BISB2 | #2, DIF\$GL_IGNORE | | 0801 |
| | | | 0158 | C6 | 9F | 0036B | 21\$: | PUSHAB | SD_ASCII | 0807 |
| 67 | | 68 | | 01 | FB | 0036F | CALLS | #1, CLISPRESENT | | |
| | 01 | 00 | | 50 | FO | 00372 | INSV | R0, #0, #1, DIF\$GL_FLAGS | | |
| | | | 016C | C6 | 9F | 00377 | PUSHAB | SD_HEXADECEMAL | | 0808 |
| 67 | | 68 | | 01 | FB | 0037B | CALLS | #1, CLISPRESENT | | |
| | 01 | 01 | | 50 | FO | 0037E | INSV | R0, #1, #1, DIF\$GL_FLAGS | | |
| | | | 017C | C6 | 9F | 00383 | PUSHAB | SD_OCTAL | | 0809 |
| 67 | | 68 | | 01 | FB | 00387 | CALLS | #1, CLISPRESENT | | |
| | 01 | 02 | | 50 | FO | 0038A | INSV | R0, #2, #1, DIF\$GL_FLAGS | | |
| | | | 68 | A6 | 9F | 0038F | PUSHAB | SD_OUTPUT | | 0814 |
| 01 | A7 | 68 | | 01 | FB | 00392 | CALLS | #1, CLISPRESENT | | |
| | | 07 | | 50 | FO | 00395 | INSV | R0, #7, #1, DIF\$GL_FLAGS+1 | | |
| | | 0E | | 50 | E9 | 0039B | BLBC | R0, 22\$ | | |
| | | | 00000000G | 00 | 9F | 0039E | PUSHAB | DIF\$GL_OUTDESC | | 0815 |
| | | | 68 | A6 | 9F | 003A4 | PUSHAB | SD_OUTPUT | | |
| | | 69 | | 02 | FB | 003A7 | CALLS | #2, CLISGET_VALUE | | |
| | | | | 22 | 11 | 003AA | BRB | 23\$ | | |
| | | 1E | 01 | A7 | EB | 003AC | 22\$: | BLBS | DIF\$GL_FLAGS+1, 23\$ | 0816 |
| | | | 00000000G | 00 | 9F | 003B0 | PUSHAB | DIF\$GL_OUTDESC | | 0817 |
| | | | 00000000' | EF | 9F | 003B6 | PUSHAB | SY\$OUTPUT+1 | | |
| 08 | AE | | 00000000' | EF | 9A | 003BC | MOVZBL | SY\$OUTPUT, 8(SP) | | |
| | | | 08 | AE | 9F | 003C4 | PUSHAB | 8(SP) | | |
| | | 00000000G | | 00 | 03 | FB | 003C7 | CALLS | #3, LIB\$SCOPY_R_DX | |
| | | | 00000000' | EF | 9F | 003CE | 23\$: | PUSHAB | FIND_MASFIL | 0823 |
| | | | 9C | A6 | 9F | 003D4 | PUSHAB | SD_INPUT1 | | |
| | | 69 | | 02 | FB | 003D7 | CALLS | #2, CLISGET_VALUE | | |
| | | | 00000000G | 00 | 9F | 003DA | PUSHAB | DIF\$GL_MASFDB | | 0824 |
| 15 | AA | | | 01 | FB | 003E0 | CALLS | #1, GETCHBARQUAL | | |
| | | | 28 | AE | 9F | 003E4 | PUSHAB | DESC | | 0829 |
| | | | 9C | A6 | 9F | 003E7 | PUSHAB | SD_INPUT1 | | |
| | | 69 | | 02 | FB | 003EA | CALLS | #2, CLISGET_VALUE | | |
| | | 52 | | 50 | DO | 003ED | MOVL | R0, STATUS | | |
| | | 0A | | 52 | E9 | 003F0 | BLBC | STATUS, 24\$ | | |
| | | | 00000000G | 00 | 9F | 003F3 | PUSHAB | DIF\$GL_REVFDB | | 0830 |
| 15 | AA | | | 01 | FB | 003F9 | CALLS | #1, GETCHBARQUAL | | |
| | | | 00000000G | 00 | 9F | 003FD | 24\$: | PUSHAB | DIF\$GL_REVDESC | 0831 |
| | | | AC | A6 | 9F | 00403 | PUSHAB | SD_INPOT2 | | |
| | | 69 | | 02 | FB | 00406 | CALLS | #2, CLISGET_VALUE | | |

| | | | | | | | | | | | | |
|--|--|----|-----------|-----------|-----------|-------|--------|---------------------------|-------------------------|---------------------------------|-----------------------|------|
| | | 12 | | 50 | E9 | 00409 | BLBC | R0, 25\$ | | | | |
| | | 1B | | 52 | E9 | 0040C | BLBC | STATUS, 26\$ | | 0833 | | |
| | | | 00000000G | 8F | DD | 0040F | PUSHL | #DIF\$ FOOMNYFIL | | 0834 | | |
| | | 00 | | 01 | FB | 00415 | CALLS | #1, LIB\$STOP | | | | |
| | | | | 0C | 11 | 0041C | BRB | 26\$ | | 0831 | | |
| | | 09 | | 52 | E9 | 0041E | 25\$: | BLBC | STATUS, 26\$ | 0836 | | |
| | | 28 | AE | 08 | 28 | 00421 | MOVC3 | #8, DESC, DIF\$GL_REVDESC | | 0837 | | |
| | | 00 | 00000000G | 00 | 9F | 0042A | 26\$: | PUSHAB | DIF\$GL_REVFDB | 0838 | | |
| | | 15 | AA | 01 | FB | 00430 | CALLS | #1, GETCHBARQUAL | | | | |
| | | | 00000000G | 00 | 9F | 00434 | PUSHAB | DIF\$GL_CMDESC | | 0843 | | |
| | | | | 8C | A6 | 9F | 0043A | PUSHAB | SD \$LINE | | | |
| | | 03 | | 69 | 02 | FB | 0043D | CALLS | #2, CLISGET VALUE | | | |
| | | | | 67 | 06 | E1 | 00440 | BBC | #6, DIF\$GL_FLAGS, 27\$ | 0848 | | |
| | | | | 67 | 01 | 88 | 00444 | BISB2 | #1, DIF\$GL_FLAGS | 0849 | | |
| | | | | 0E | 01 | A7 | E9 | 00447 | 27\$: | BLBC | DIF\$GL_FLAGS+1, 28\$ | 0851 |
| | | | | 6B | 40 | 8F | 88 | 0044B | BISB2 | #64, DIF\$GL_IGNORE | 0853 | |
| | | | | 6B | 80 | 8F | 8A | 0044F | BICB2 | #128, DIF\$GL_IGNORE | 0854 | |
| | | 01 | A7 | 10 | 8A | 00453 | BICB2 | #16, DIF\$GL_FLAGS+1 | | 0855 | | |
| | | | | 19 | 11 | 00457 | BRB | 29\$ | | 0851 | | |
| | | | | 67 | 95 | 00459 | 28\$: | TSTB | DIF\$GL_FLAGS | 0857 | | |
| | | | | 15 | 19 | 0045B | BLSS | 29\$ | | | | |
| | | | 0E | 00000000G | 00 | E8 | 0045D | BLBS | DIF\$GL_MASFDB+36, 29\$ | 0858 | | |
| | | | 07 | 00000000G | 00 | E8 | 00464 | BLBS | DIF\$GL_REVFDB+36, 29\$ | 0859 | | |
| | | 03 | | 67 | 06 | E0 | 0046B | BBS | #6, DIF\$GL_FLAGS, 29\$ | 0860 | | |
| | | | | 67 | 20 | 88 | 0046F | BISB2 | #32, DIF\$GL_FLAGS | 0861 | | |
| | | | | 51 | 00000000G | 00 | D0 | 00472 | 29\$: | MOVL | DIF\$GL_MERGED, R1 | 0865 |
| | | | | 50 | 00000000G | 00 | D0 | 00479 | MOVL | DIF\$GL_MATCH, R0 | | |
| | | | | 50 | 51 | D1 | 00480 | CMPL | R1, R0 | | | |
| | | | | | 1E | 15 | 00483 | BLEQ | 30\$ | | | |
| | | | | | 50 | DD | 00485 | PUSHL | R0 | | 0867 | |
| | | | | | 51 | DD | 00487 | PUSHL | R1 | | | |
| | | | | | 02 | DD | 00489 | PUSHL | #2 | | | |
| | | | | | 8F | DD | 0048B | PUSHL | #DIF\$ MRGGTMTCH | | | |
| | | | 00000000G | 00 | 04 | FB | 00491 | CALLS | #4, LIB\$SIGNAL | | | |
| | | | 00000000G | 00 | 00000000G | 00 | D0 | 00498 | MOVL | DIF\$GL_MATCH, DIF\$GL_MERGED | 0868 | |
| | | | | 51 | 00000000G | 00 | D0 | 004A3 | 30\$: | MOVL | DIF\$GL_PARALLEL, R1 | 0871 |
| | | | | 50 | 00000000G | 00 | D0 | 004AA | MOVL | DIF\$GL_MATCH, R0 | | |
| | | | | 50 | 51 | D1 | 004B1 | CMPL | R1, R0 | | | |
| | | | | | 1E | 15 | 004B4 | BLEQ | 31\$ | | | |
| | | | | | 50 | DD | 004B6 | PUSHL | R0 | | 0873 | |
| | | | | | 51 | DD | 004B8 | PUSHL | R1 | | | |
| | | | | | 02 | DD | 004BA | PUSHL | #2 | | | |
| | | | | | 8F | DD | 004BC | PUSHL | #DIF\$ PARGTMTCH | | | |
| | | | 00000000G | 00 | 04 | FB | 004C2 | CALLS | #4, LIB\$SIGNAL | | | |
| | | | 00000000G | 00 | 00000000G | 00 | D0 | 004C9 | MOVL | DIF\$GL_MATCH, DIF\$GL_PARALLEL | 0874 | |
| | | | | 50 | 01 | D0 | 004D4 | 31\$: | MOVL | #1, R0 | 0877 | |
| | | | | | 04 | 004D7 | RET | | | 0878 | | |

; Routine Size: 1240 bytes, Routine Base: \$CODE\$ + 00F4

```

584 0879 1 ROUTINE getfilename (fab) =
585 0880 2 BEGIN
586 0881 2 !**
587 0882 2 FUNCTIONAL DESCRIPTION:
588 0883 2
589 0884 2 This routine returns a string descriptor for a file.
590 0885 2
591 0886 2 Inputs:
592 0887 2
593 0888 2 fab Address of the fab
594 0889 2
595 0890 2 Outputs:
596 0891 2
597 0892 2 Routine value is address of string descriptor for file name
598 0893 2
599 0894 2 --
600 0895 2
601 0896 2 MAP
602 0897 2 fab : REF BBLOCK;
603 0898 2
604 0899 2 BIND
605 0900 2 nam = .fab [fab$l_nam] : BBLOCK;
606 0901 2
607 0902 2 OWN
608 0903 2 filedesc : BBLOCK[dsc$sc_s_bln];
609 0904 2
610 0905 2 IF (filedesc [dsc$w_length] = .nam [nam$b_rsl]) NEQ 0 !If resultant name present
611 0906 2 THEN filedesc [dsc$a_pointer] = .nam [nam$l_rsa]
612 0907 2 ELSE IF (filedesc [dsc$w_length] = .nam [nam$b_est]) NEQ 0 !If expanded name present
613 0908 2 THEN filedesc [dsc$a_pointer] = .nam [nam$t_esa]
614 0909 2 ELSE BEGIN
615 0910 2 filedesc [dsc$w_length] = .fab [fab$b_fns]; !Use filename string
616 0911 2 filedesc [dsc$a_pointer] = .fab [fab$t_fna]; ! if all else fails
617 0912 2 END;
618 0913 2
619 0914 2 RETURN filedesc
620 0915 1 END; !Of getfilename

```

.PSECT \$OWNS,NOEXE,2

0004 FILEDESC:
.BLKB 8

.PSECT \$CODES,NOVRT,2

0004 0000 GETFILENAME:

| | | | | | | | |
|----|-----------|----|----|-------|---------|-----------------|-------------------|
| | | | | .WORD | Save R2 | : 0879 | |
| 52 | 00000000' | EF | 9E | 00002 | MOVAB | FILEDESC, R2 | |
| 51 | 04 | AC | D0 | 00009 | MOVL | FAB, R1 | |
| 50 | 28 | A1 | D0 | 0000D | MOVL | 40(R1), R0 | |
| 62 | 03 | A0 | 9B | 00011 | MOVZBW | 3(R0), FILEDESC | |
| | | | 07 | 13 | BEQL | 1\$ | |
| 04 | A2 | 04 | A0 | D0 | 00017 | MOVL | 4(R0), FILEDESC+4 |
| | | | | | | : 0906 | |

DIF_GETCMD
V04=000

DIFSGETCMD Process command line

M 14
15-Sep-1984 23:39:15
14-Sep-1984 12:19:20

VAX-11 Bliss-32 V4.0-742
[DIF.SRC]DIFGETCMD.B32;1

Page 30
(9)

| | | | | | | | | | | |
|----|----|----|----|-------|-------|------|--------|--------------------|--|--------|
| | 62 | 0B | 16 | 11 | U001C | | BRB | 3\$ | | : |
| | | | A0 | 9B | 0001E | 1\$: | MOVZBW | 11(R0), FILEDESC | | : 0907 |
| | | | 07 | 13 | 00022 | | BEQL | 2\$ | | : |
| 04 | A2 | 0C | A0 | D0 | 00024 | | MOVL | 12(R0), FILEDESC+4 | | : 0908 |
| | | | 09 | 11 | 00029 | | BRB | 3\$ | | : |
| | 62 | 34 | A1 | 9B | 0002B | 2\$: | MOVZBW | 52(R1), FILEDESC | | : 0910 |
| 04 | A2 | 2C | A1 | D0 | 0002F | | MOVL | 44(R1), FILEDESC+4 | | : 0911 |
| | 50 | | 62 | 9E | 00034 | 3\$: | MOVAB | FILEDESC, R0 | | : 0914 |
| | | | 04 | 00037 | | | RET | | | : 0915 |

; Routine Size: 56 bytes, Routine Base: \$CODE\$ + 05CC

DIF\$GETCMD Process command line

```

622 0916 1 GLOBAL ROUTINE dif$open_mas =
623 0917 2 BEGIN
624 0918 3
625 0919 4   Open the master input file
626 0920 5
627 0921 6 LOCAL
628 0922 7   xabfhc : $XABFHC_DECL,
629 0923 8   status,
630 0924 9   strstatus;
631 0925 10
632 0926 11 status = lib$find_file(find_masfile,
633 0927 12   dif$gl_masdesc, find_context,
634 0928 13   0, dif$gl_masdesc);           !No default file spec
635 0929 14 IF NOT .status
636 0930 15 THEN BEGIN
637 0931 16   IF .status EQL rms$nmf
638 0932 17   THEN status = rms$fnf;
639 0933 18   SIGNAL(dif$_openin, 1, dif$gl_masdesc,
640 0934 19   .status,
641 0935 20   .find_context[fab$l_stv]);
642 0936 21 END;
643 0937 22
644 0938 23 Set pointer in FDB to NAM block
645 0939 24
646 0940 25 IF .dif$gl_masnam EQL 0
647 0941 26 THEN BEGIN
648 0942 27   dif$gl_masnam = .find_context[fab$l_nam];
649 0943 28   dif$gl_masfdb[fdb$l_namptr] = .find_context[fab$l_nam];
650 0944 29 END;
651 0945 30
652 0946 31 IF .dif$gl_masnam[nam$w_wildcard]
653 0947 32 THEN SIGNAL_STOP(dif$_nowild);
654 0948 33
655 0949 34 initfile(dif$gl_masfdb, 0, .find_context);
656 0950 35
657 0951 36
658 0952 37 Open the file
659 0953 38
660 P 0954 39 $XABFHC_INIT(xab=xabfhc,
661 0955 40   nxt=0);
662 0956 41 find_context[fab$l_xab] = xabfhc;
663 0957 42 status = $OPEN(fab=.find_context);
664 0958 43 strstatus = str$copy_dx(dif$gl_masdesc, getfilename(.find_context));
665 0959 44 IF NOT .strstatus
666 0960 45 THEN SIGNAL_STOP(dif$_strcopyerr, 0, .strstatus);
667 0961 46 IF NOT .status
668 0962 47 THEN SIGNAL_STOP(dif$_openin, 1, dif$gl_masdesc,
669 0963 48   .status, .find_context[fab$l_stv]);
670 0964 49
671 0965 50 Save the IFI to close the file
672 0966 51
673 0967 52 dif$gl_masfdb[fdb$w_ifi] = .find_context[fab$w_ifi];
674 0968 53
675 0969 54 Connect the record stream
676 0970 55
677 0971 56 status = $CONNECT(rab=dif$gl_masrab);
678 0972 57 IF NOT .status
```


| | | | | | | | | | | | |
|----|-----------|----|-----------|----|----|-------|------|--------|------------------------------|--|------|
| | | | 0C | A0 | DD | 00053 | | PUSHL | 12(R0) | | |
| | | | | 56 | DD | 00056 | | PUSHL | STATUS | | 0934 |
| | | | | 58 | DD | 00058 | | PUSHL | R8 | | 0933 |
| | | | | 01 | DD | 0005A | | PUSHL | #1 | | |
| | | | | 5B | DD | 0005C | | PUSHL | R11 | | |
| | 00000000G | 00 | | 05 | FB | 0005E | | CALLS | #5, LIB\$SIGNAL | | |
| | | | | 6A | D5 | 0C065 | 2\$: | TSTL | DIF\$GL_MASNAM | | 0940 |
| | | | | 0F | I2 | 00067 | | BNEQ | 3\$ | | |
| | | 50 | | 67 | DD | 00069 | | MOVL | FIND_CONTEXT, R0 | | 0942 |
| | | 6A | 28 | A0 | DD | 0006C | | MOVL | 40(R0), DIF\$GL_MASNAM | | |
| | 00000000G | 00 | 28 | A0 | DD | 00070 | | MOVL | 40(R0), DIF\$GL_MASFDB+52 | | 0943 |
| | | 50 | | 6A | DD | 00078 | 3\$: | MOVL | DIF\$GL_MASNAM, R0 | | 0946 |
| | | 09 | 35 | A0 | E9 | 0007B | | BLBC | 53(R0), 4\$ | | |
| | | | 00000000G | 8F | DD | 0007F | | PUSHL | #DIF\$ NOWILD | | 0947 |
| | | 69 | | 01 | FB | 00085 | | CALLS | #1, LIB\$STOP | | |
| | | | | 67 | DD | 00088 | 4\$: | PUSHL | FIND_CONTEXT | | 0949 |
| | | | | 7E | D4 | 0008A | | CLRL | -(SP) | | |
| | | | 00000000G | 00 | 9F | 0008C | | PUSHAB | DIF\$GL_MASFDB | | |
| | 00000000V | EF | | 03 | FB | 00092 | | CALLS | #3, INITFILE | | |
| 2C | 00 | 6E | | 00 | 2C | 00099 | | M-1C5 | #0, (SP), #0, #44, \$RMS_PTR | | 0955 |
| | | | 04 | AE | | 0009E | | | | | |
| | 04 | AE | 2C1D | 8F | B0 | 000A0 | | MOVW | #11293, \$RMS_PTR | | |
| | | 50 | | 67 | DD | 000A6 | | MOVL | FIND_CONTEXT, R0 | | 0956 |
| | 24 | A0 | 04 | AE | 9E | 000A9 | | MOVAB | XABFRC, 36(R0) | | |
| | | | | 50 | DD | 000AE | | PUSHL | R0 | | 0957 |
| | 00000000G | 00 | | 01 | FB | 000B0 | | CALLS | #1, SYSSOPEN | | |
| | | 56 | | 50 | DD | 000B7 | | MOVL | R0, STATUS | | |
| | | | | 67 | DD | 000BA | | PUSHL | FIND_CONTEXT | | 0958 |
| | FF07 | CF | | 01 | FB | 000BC | | CALLS | #1, GETFILENAME | | |
| | | | | 50 | DD | 000C1 | | PUSHL | R0 | | |
| | | | | 58 | DD | 000C3 | | PUSHL | R8 | | |
| | 00000000G | 00 | | 02 | FB | 000C5 | | CALLS | #2, STR\$COPY DX | | |
| | | 54 | | 50 | DD | 000CC | | MOVL | R0, STRSTATUS | | |
| | | 0D | | 54 | E8 | 000CF | | BLBS | STRSTATUS, 5\$ | | 0959 |
| | | | | 54 | DD | 000D2 | | PUSHL | STRSTATUS | | 0960 |
| | | | | 7E | D4 | 000D4 | | CLRL | -(SP) | | |
| | | | 00000000G | 8F | DD | 000D6 | | PUSHL | #DIF\$ STRCPYERR | | |
| | | 69 | | 03 | FB | 000DC | | CALLS | #3, LIB\$STOP | | |
| | | 11 | | 56 | E8 | 000DF | 5\$: | BLBS | STATUS, 6\$ | | 0961 |
| | | 50 | | 67 | DD | 000E2 | | MOVL | FIND_CONTEXT, R0 | | 0963 |
| | | | 0C | A0 | DD | 000E5 | | PUSHL | 12(R0) | | |
| | | | | 56 | DD | 000E8 | | PUSHL | STATUS | | |
| | | | | 58 | DD | 000EA | | PUSHL | R8 | | 0962 |
| | | | | 01 | DD | 000EC | | PUSHL | #1 | | |
| | | | | 5B | DD | 000EE | | PUSHL | R11 | | |
| | | 69 | | 05 | FB | 000F0 | | CALLS | #5, LIB\$STOP | | |
| | | 50 | | 67 | DD | 000F3 | 6\$: | MOVL | FIND_CONTEXT, R0 | | 0967 |
| | 00000000G | 00 | 02 | A0 | B0 | 000F6 | | MOVW | 2(R0), DIF\$GL_MASFDB+44 | | |
| | | | 00000000G | 00 | 9F | 000FE | | PUSHAB | DIF\$GL_MASRAB | | 0971 |
| | 00000000G | 00 | | 01 | FB | 00104 | | CALLS | #1, SYSSCONNECT | | |
| | | 56 | | 50 | DD | 0010B | | MOVL | R0, STATUS | | |
| | | 11 | | 56 | E8 | 0010E | | BLBS | STATUS, 7\$ | | 0972 |
| | | | 00000000G | 00 | DD | 00111 | | PUSHL | DIF\$GL_MASRAB+12 | | 0974 |
| | | | | 56 | DD | 00117 | | PUSHL | STATUS | | |
| | | | | 58 | DD | 00119 | | PUSHL | R8 | | 0973 |
| | | | | 01 | DD | 0011B | | PUSHL | #1 | | |
| | | | | 5B | DD | 0011D | | PUSHL | R11 | | |

| | | | | | | | | | | |
|--------------|-----------|-----------|----|-------|-------------|------------------------|-------------------------|----------------------------|------|--|
| 69 | 00000000G | 05 | FB | 0011F | CALLS | #5, LIB\$STOP | : | | | |
| | 08 | 00 | 9F | 00122 | 7\$: PUSHAB | DIF\$GL_MASRAB | : | 0978 | | |
| | | AE | 9F | 00128 | PUSHAB | XABFHC | : | | | |
| 00000000V | EF | 02 | FB | 0012B | CALLS | #2, ALLOCATE_UBF | : | | | |
| | 50 | 67 | D0 | 00132 | MOVL | FIND_CONTEXT, R0 | : | 0979 | | |
| | | 24 | A0 | 00135 | CLRL | 36(R0) | : | | | |
| 08 00000000G | 00 | 03 | E1 | 00138 | BBC | #3, DIF\$GL_FLAGS, 8\$ | : | 0984 | | |
| | | 00000000G | 00 | B5 | 00140 | TSTW | DIF\$GL_COMMDESC | : | | |
| | | | 69 | 12 | 00146 | BNEQ | 13\$ | : | | |
| | | | 56 | D4 | 00148 | 8\$: CLRL | I | : | 0985 | |
| | | | 5C | 11 | 0014A | BRB | 12\$ | : | | |
| 50 | 56 | 0A | C5 | 0014C | 9\$: MULL3 | #10, I, R0 | : | 0988 | | |
| | 55 | 00000000G | 00 | 40 | 9E | 00150 | MOVAB | DIF\$GL_TYPCDATA[R0], R5 | : | |
| | 51 | | 65 | D0 | 00158 | MOVL | (R5), R1 | : | 0989 | |
| | 50 | | 6A | D0 | 0015B | MOVL | DIF\$GL_MASNAM, R0 | : | 0992 | |
| | 53 | 3C | A0 | 9A | 0015E | MOVZBL | 60(R0), R3 | : | | |
| | 52 | | 61 | 9A | 00162 | MOVZBL | (R1), R2 | : | 0993 | |
| 52 | 00 | 50 | B0 | 53 | 2D | 00165 | CMPCS | R3, @80(R0), #0, R2, 1(R1) | : | |
| | | | 01 | A1 | | 0016B | | | : | |
| | | | 37 | 12 | 0016D | BNEQ | 11\$ | : | | |
| | | 00000000G | 00 | 9F | 0016F | PUSHAB | DIF\$GL_COMMDESC | : | 0995 | |
| 7E | 04 | A5 | 01 | C1 | 00175 | ADDL3 | #1, 4(R5), -(SP) | : | | |
| | 08 | AE | 04 | B5 | 9A | 0017A | MOVZBL | @4(R5), 8(SP) | : | |
| | | | 08 | AE | 9F | 0017F | PUSHAB | 8(SP) | : | |
| 00000000G | 00 | | 03 | FB | 00182 | CALLS | #3, LIB\$SCOPY_R_DX | : | | |
| | 54 | | 50 | D0 | 00189 | MOVL | R0, STRSTATUS | : | | |
| | 0D | | 54 | E8 | 0018C | BLBS | STRSTATUS, 10\$ | : | | |
| | | | 54 | DD | 0018F | PUSHL | STRSTATUS | : | 0997 | |
| | | | 7E | D4 | 00191 | CLRL | -(SP) | : | | |
| | | 00000000G | 8F | DD | 00193 | PUSHL | #DIF\$_STRCPYERR | : | | |
| | 69 | | 03 | FB | 00199 | CALLS | #3, LIB\$STOP | : | | |
| 00000000G | 00 | 08 | A5 | 3C | 0019C | 10\$: MOVZWL | 8(R5), DIF\$GL_COMMFLGS | : | 0998 | |
| | | | 08 | 11 | 001A4 | BRB | 13\$ | : | 0994 | |
| | | | 56 | D6 | 001A6 | 11\$: INCL | I | : | 0985 | |
| 00000000G | 8F | | 56 | D1 | 001A8 | 12\$: CMPL | I, #DIF\$C_MAXLANGS-1 | : | | |
| | | | 9B | 1B | 001AF | BLEQU | 9\$ | : | | |
| | 50 | | 01 | D0 | 001B1 | 13\$: MOVL | #1, R0 | : | 1003 | |
| | | | 04 | 001B4 | RET | | : | 1004 | | |

; Routine Size: 437 bytes, Routine Base: \$CODE\$ + 0604

DIF\$GETCMD Process command line

```
712 1005 1 GLOBAL ROUTINE dif$open_rev =
713 1006 2 BEGIN
714 1007 3
715 1008 4   Open the revision file
716 1009 5
717 1010 6 LOCAL
718 1011 7   status,
719 1012 8   strstatus,
720 1013 9   xabfhc : $XABFHC_DECL,
721 1014 10  fab : $FAB_DECL;
722 1015 11
723 1016 12   initfile(dif$gl_revfdb,dif$gl_revrrs,fab);
724 1017 13 P $XABFHC_INIT(xab=xabfhc,
725 1018 14   nxt=0);
726 1019 15   fab[fab$l_xab] = xabfhc;
727 1020 16
728 1021 17   Set the master file nam block as the related file
729 1022 18
730 1023 19   dif$gl_revnam[nam$l_rlf] = .dif$gl_masnam;
731 1024 20
732 1025 21   See if there is a revision file specified. If not,
733 1026 22   then look for the next lower version of the master file
734 1027 23
735 1028 24 IF .fab[fab$b_fns] EQL 0
736 1029 25 THEN BEGIN
737 1030 26 LOCAL
738 1031 27   verdesc : desc_decl,
739 1032 28   file_length,
740 1033 29   verlength,
741 1034 30   version,
742 1035 31   namrss : BBLOCK[nam$c_maxrss],
743 1036 32   verptr : REF VECTOR[,BYTE];
744 1037 33
745 1038 34   Copy all of the master file spec except the file version field
746 1039 35
747 1040 36   verptr = CHSMOVE((.dif$gl_masnam[nam$l_type]-
748 1041 37   .dif$gl_masnam[nam$l_rsa]+.dif$gl_masnam[nam$b_type]+1),
749 1042 38   .dif$gl_masnam[nam$l_rsa],namrss);
750 1043 39   file_length = .verptr - namrss;
751 1044 40   verdesc[dsc$w_length] = nam$c_maxrss - .file_length;
752 1045 41   verdesc[dsc$a_pointer] = .verptr;
753 1046 42
754 1047 43   Convert master file version number to binary
755 1048 44
756 1049 45   lib$cvt_dtb(.dif$gl_masnam[nam$b_ver] - 1,
757 1050 46   .dif$gl_masnam[nam$l_ver] + 1,version);
758 1051 47   verlength = 0;
759 1052 48
760 1053 49   Now loop, looking for a lower version of the master file
761 1054 50
762 1055 51 WHILE (version = .version - 1) GTR 0
763 1056 52 DO BEGIN
764 1057 53   status = SY$FAO($descriptor('!UL'),verlength,verdesc,.version);
765 1058 54   IF NOT .status
766 1059 55   THEN SIGNAL_STOP(dif$sysfaoerr,0,.status);
767 1060 56   fab[fab$b_fns] = .file_length + .verlength;
768 1061 57   fab[fab$l_fna] = namrss;
```

```

: 769      1062 5      IF (status = $OPEN(fab=fab))
: 770      1063 4          THEN EXITLOOP
: 771      1064 4          ELSE IF .status NEQ rms$_fnf
: 772      1065 4              THEN EXITLOOP;
: 773      1066          END;
: 774      1067          IF .version EQL 0                      !Did we find one?
: 775      1068              THEN SIGNAL_STOP(dif$_nolower,1,dif$gl_masdesc);
: 776      1069          END
: 777      1070          ELSE status = $OPEN(fab=fab);
: 778      1071
: 779      1072          strstatus = strscopy_dx(dif$gl_revdesc,getfilename(fab));
: 780      1073
: 781      1074          IF NOT .strstatus
: 782      1075              THEN SIGNAL_STOP(dif$_strcpyerr,0,.strstatus);
: 783      1076          IF NOT .status
: 784      1077              THEN SIGNAL_STOP(dif$_openin,1,dif$gl_revdesc,
: 785      1078                  .status,.fab[fab$_stv]);
: 786      1079
: 787      1080          ! Save the IFI for $CLOSE
: 788      1081
: 789      1082          dif$gl_revfdb[fdb$_w_ifi] = .fab[fab$_w_ifi];
: 790      1083
: 791      1084          ! Connect the record stream
: 792      1085
: 793      1086          status = $CONNECT(rab=dif$gl_revrab);
: 794      1087          IF NOT .status
: 795      1088              THEN SIGNAL_STOP(dif$_openin,1,dif$gl_revdesc,
: 796      1089                  .status,.dif$gl_revrab[rab$_stv]);
: 797      1090
: 798      1091          ! Allocate user buffer
: 799      1092
: 800      1093          allocate_ubf(xabfhc,dif$gl_revrab);
: 801      1094
: 802      1095          RETURN true
: 803      1096          END;

```

.PSECT \$SPLITS,NOWRT,NOEXE,2

```

4C 55 21 00220 P.ACL: .ASCII \!UL\
          00223      .BLKB 1
00000003 00224 P.ACK: .LONG 3
00000000' 00228      .ADDRESS P.ACL

```

.PSECT \$CODE\$,NOWRT,2

```

OFFC 00000      .ENTRY DIF$OPEN_REV, Save R2,R3,R4,R5,R6,R7,R8,R9,-; 1005
5B 00000000G 8F D0 00002      MOVL #DIF$_OPENIN, R11
5A 00000000G 00 9F 00009      MOVAB SYSSOPEN, R10
59 00000000G 00 9E 00010      MOVAB DIF$GL_REVRAB, R9
58 00000000G 00 9E 00017      MOVAB DIF$GL_REVDESC, R8
57 00000000G 00 9E 0001E      MOVAB LIB$STOP, R7
5E FE74 CE 9E 00025      MOVAB -396(SP), SP

```

| | | | | | | | | | | | |
|------|----|-----------|----|-----------|-----------|-------|----------|------|--------|--------------------------------|------|
| | | | | 84 | AD | 9F | 0002A | | PUSHAB | FAB | 1016 |
| | | | | 00000000G | 00 | 9F | 0002D | | PUSHAB | DIF\$GL_REVRSS | |
| | | | | 00000000G | 00 | 9F | 00033 | | PUSHAB | DIF\$GL_REVFDB | |
| 2C | 00 | 00000000V | EF | | 03 | FB | 00039 | | CALLS | #3, INITFILE | |
| | | | 6E | | 00 | 2C | 00040 | | MOVCS | #0, (SP), #0, #44, \$RMS_PTR | 1018 |
| | | | | D4 | AD | | 00045 | | | | |
| | | D4 | AD | 2C1D | 8F | B0 | 00047 | | MOVW | #11293, \$RMS_PTR | |
| | | AB | AD | D4 | AD | 9E | 0004D | | MOVAB | XABFHC, FAB+36 | 1019 |
| | | | 56 | 00000000G | 00 | D0 | 00052 | | MOVL | DIF\$GL_MASNAM, R6 | 1023 |
| | | 00000000G | 00 | | 56 | D0 | 00059 | | MOVL | R6, DIF\$GL_REVNAM+16 | |
| | | | | B8 | AD | 95 | 00060 | | TSTB | FAB+52 | 1028 |
| | | | | | 03 | 13 | 00063 | | BEQL | 1\$ | |
| | | | | 00A4 | 31 | 00065 | | | BRW | 5\$ | |
| | 50 | 50 | A6 | 04 | A6 | C3 | 00068 | 1\$: | SUBL3 | 4(R6), 80(R6), R0 | 1041 |
| | | | 51 | 3C | A6 | 9A | 0006E | | MOVZBL | 60(R6), R1 | |
| | | | 50 | 01 | A140 | 9E | 00072 | | MOVAB | 1(R1)[R0], R0 | 1040 |
| 08 | AE | 04 | B6 | | 50 | 28 | 00077 | | MOVCS | R0, 24(R6), NAMRSS | |
| | | | 50 | | 08 | AE | 0007D | | MOVAB | NAMRSS, R0 | 1043 |
| | | | 53 | | 50 | C3 | 00081 | | SUBL3 | R0, VERPTR, FILE_LENGTH | |
| FF7C | CD | 00FF | 8F | | 54 | A3 | 00085 | | SUBW3 | FILE_LENGTH, #255, VERDESC | 1044 |
| | | 80 | AD | | 53 | D0 | 0008D | | MOVL | VERPTR, VERDESC+4 | 1045 |
| | | | | | 5E | DD | 00091 | | PUSHL | SP | 1049 |
| | 7E | 54 | A6 | | 01 | C1 | 00093 | | ADDL3 | #1, 84(R6), -(SP) | 1050 |
| | | | 7E | | 3D | A6 | 00098 | | MOVZBL | 61(R6), -(SP) | 1049 |
| | | | | | 6E | D7 | 0009C | | DECL | (SP) | |
| | | 00000000G | 00 | | 03 | FB | 0009E | | CALLS | #3, LIB\$CVT_DTB | |
| | | | | | 04 | AE | D4 000A5 | | CLRL | VERLENGTH | 1051 |
| | | | | | 6E | D7 | 000AB | 2\$: | DECL | VERSION | 1055 |
| | | | | | 49 | 15 | 000AA | | BLEQ | 4\$ | |
| | | | | | 6E | DD | 000AC | | PUSHL | VERSION | 1057 |
| | | | | FF7C | CD | 9F | 000AE | | PUSHAB | VERDESC | |
| | | | | 0C | AE | 9F | 000B2 | | PUSHAB | VERLENGTH | |
| | | | | 00000000G | 00 | EF | 9F 000B5 | | PUSHAB | P.ACK | |
| | | | | | 04 | FB | 000BB | | CALLS | #4, SYSS\$FAO | |
| | | | | | 50 | D0 | 000C2 | | MOVL | R0, STATUS | |
| | | | | | 52 | E8 | 000C5 | | BLBS | STATUS, 3\$ | 1058 |
| | | | | | 52 | DD | 000C8 | | PUSHL | STATUS | 1059 |
| | | | | | 7E | D4 | 000CA | | CLRL | -(SP) | |
| | | | | 00000000G | 8F | DD | 000CC | | PUSHL | #DIF\$_SYSFAOERR | |
| | | | | | 03 | FB | 000D2 | | CALLS | #3, LIB\$STOP | |
| B8 | AD | | 54 | 04 | AE | 81 | 000D5 | 3\$: | ADDB3 | VERLENGTH, FILE_LENGTH, FAB+52 | 1060 |
| | | | | | 08 | AE | 9E 000DB | | MOVAB | NAMRSS, FAB+44 | 1061 |
| | | | B0 | AD | 84 | AD | 9F 000E0 | | PUSHAB | FAB | 1062 |
| | | | | | 01 | FB | 000E3 | | CALLS | #1, SYSS\$OPEN | |
| | | | | | 50 | D0 | 000E6 | | MOVL | R0, STATUS | |
| | | | | | 52 | E8 | 000E9 | | BLBS | STATUS, 4\$ | |
| | | 00018292 | | | 52 | D1 | 000EC | | CMPL | STATUS, #98962 | 1064 |
| | | | | | B3 | 13 | 000F3 | | BEQL | 2\$ | |
| | | | | | 6E | D5 | 000F5 | 4\$: | TSTL | VERSION | 1067 |
| | | | | | 1C | 12 | 000F7 | | BNEQ | 6\$ | |
| | | | | | 00000000G | 00 | 9F 000F9 | | PUSHAB | DIF\$GL_MASDESC | 1068 |
| | | | | | 01 | DD | 000FF | | PUSHL | #1 | |
| | | | | | 00000000G | 8F | DD 00101 | | PUSHL | #DIF\$_NOLOWVER | |
| | | | 67 | | 03 | FB | 00107 | | CALLS | #3, LIB\$STOP | |
| | | | | | 09 | 11 | 0010A | | BRB | 6\$ | 1028 |
| | | | | | 84 | AD | 9F 0010C | 5\$: | PUSHAB | FAB | 1070 |
| | | | | 6A | 01 | FB | 0010F | | CALLS | #1, SYSS\$OPEN | |

| | | | | | | | | | | | |
|--|-----------|----|----|-------|-------|------|--------|-------------------|-------------------|--|------|
| | 52 | | 50 | DD | 00112 | | MOVL | R0, | STATUS | | |
| | | 84 | AD | 9F | 00115 | 6\$: | PUSHAB | FAB | | | 1072 |
| | FCF6 | CF | 01 | FB | 00118 | | CALLS | #1, | GETFILENAME | | |
| | | | 50 | DD | 0011D | | PUSHL | R0 | | | |
| | | | 58 | DD | 0011F | | PUSHL | R8 | | | |
| | 00000000G | 00 | 02 | FB | 00121 | | CALLS | #2, | STR\$COPY_DX | | |
| | | 0D | 50 | E8 | 00128 | | BLBS | STR\$STATUS, | 7\$ | | 1074 |
| | | | 50 | DD | 0012B | | PUSHL | STR\$STATUS | | | 1075 |
| | | | 7E | D4 | 0012D | | CLRL | -(SP) | | | |
| | | | 8F | DD | 0J12F | | PUSHL | #DIF\$ | STRCPYERR | | |
| | | 67 | 03 | FB | 00135 | | CALLS | #3, | LIB\$STOP | | |
| | | 0E | 52 | E8 | 00138 | 7\$: | BLBS | STATUS, | 8\$ | | 1076 |
| | | | AD | DD | 0013B | | PUSHL | FAB+12 | | | 1078 |
| | | | 52 | DD | 0013E | | PUSHL | STATUS | | | |
| | | | 58 | DD | 00140 | | PUSHL | R8 | | | 1077 |
| | | | 01 | DD | 00142 | | PUSHL | #1 | | | |
| | | | 5B | DD | 00144 | | PUSHL | R11 | | | |
| | | 67 | 05 | FB | 00146 | | CALLS | #5, | LIB\$STOP | | |
| | 00000000G | 00 | AD | B0 | 00149 | 8\$: | MOVW | FAB+2, | DIF\$GL_REVFD8+44 | | 1082 |
| | | | 59 | DD | 00151 | | PUSHL | R9 | | | 1086 |
| | 00000000G | 00 | 01 | FB | 00153 | | CALLS | #1, | SYSS\$CONNECT | | |
| | | | 50 | DD | 0015A | | MOVL | R0, | STATUS | | |
| | | | 52 | E8 | 0015D | | BLBS | STATUS, | 9\$ | | 1087 |
| | | | A9 | DD | 00160 | | PUSHL | DIF\$GL_REVRAB+12 | | | 1089 |
| | | | 52 | DD | 00163 | | PUSHL | STATUS | | | |
| | | | 58 | DD | 00165 | | PUSHL | R8 | | | 1088 |
| | | | 01 | DD | 00167 | | PUSHL | #1 | | | |
| | | | 5B | DD | 00169 | | PUSHL | R11 | | | |
| | | 67 | 05 | FB | 0016B | | CALLS | #5, | LIB\$STOP | | |
| | | | 59 | DD | 0016E | 9\$: | PUSHL | R9 | | | 1093 |
| | | | AD | 9F | 00170 | | PUSHAB | XABFHC | | | |
| | 00000000V | EF | 02 | FB | 00173 | | CALLS | #2, | ALLOCATE_UBF | | |
| | | 50 | 01 | DD | 0017A | | MOVL | #1, | R0 | | 1095 |
| | | | 04 | 0017D | | | RET | | | | 1096 |

; Routine Size: 382 bytes, Routine Base: \$CODE\$ + 0789

DIF\$GETCMD Process command line

```
805 1097 1 GLOBAL ROUTINE dif$open_out =
806 1098 2 BEGIN
807 1099 3
808 1100 4 Open the output file
809 1101 5
810 1102 6 LOCAL
811 1103 7     status,
812 1104 8     strstatus,
813 1105 9     fab : $FAB_DECL;
814 1106 10
815 1107 11
816 1108 12 Initialize the FAB and NAM blocks
817 1109 13
818 1110 14 $FAB_INIT(fab=fab,
819 1111 15     fns=.dif$gl_outdesc[dsc$w_length],
820 1112 16     fna=.dif$gl_outdesc[dsc$a_pointer],
821 1113 17     fop=ofp,
822 1114 18     dnm='.DIF',
823 1115 19     rat=cr,
824 1116 20     fac=put,
825 1117 21     nam=dif$gl_outnam);
826 1118 22
827 1119 23 $NAM_INIT(nam=dif$gl_outnam,
828 1120 24     rlf=.dif$gl_masnam,
829 1121 25     esa=dif$gl_outrss,
830 1122 26     ess=nam$c_maxrss,
831 1123 27     rsa=dif$gl_outrss,
832 1124 28     rss=nam$c_maxrss);
833 1125 29
834 1126 30 Attempt to open the output file
835 1127 31
836 1128 32 status = $CREATE(fab=fab);
837 1129 33 strstatus = str$copy_dx(dif$gl_outdesc,getfilename(fab));
838 1130 34
839 1131 35 IF NOT .strstatus
840 1132 36 THEN SIGNAL_STOP(dif$strcpyerr,0,.strstatus);
841 1133 37 IF NOT .status
842 1134 38 THEN SIGNAL_STOP(dif$openout,1,dif$gl_outdesc,
843 1135 39     .status,.fab[fab$l_stv]);
844 1136 40
845 1137 41 dif$gl_outifi = .fab[fab$w_ifi];
846 1138 42
847 1139 43
848 1140 44 Initialize the RAB and connect the record stream
849 1141 45
850 1142 46 $RAB_INIT(rab=dif$gl_outrab,
851 1143 47     rac=seq,
852 1144 48     fab=fab);
853 1145 49
854 1146 50 IF NOT (status = $CONNECT(rab=dif$gl_outrab))
855 1147 51 THEN SIGNAL_STOP(dif$openout,1,dif$gl_outdesc,
856 1148 52     .status,.dif$gl_outrab[rab$l_stv]);
857 1149 53
858 1150 54
859 1151 55 If no /LINE_WIDTH specified, determine width of device
860 1152 56
861 1153 57 IF NOT .dif$gl_flags[dif$v_width]
```

```

: 862      1154 2      THEN find_list_width(fab);
: 863      1155 2
: 864      1156 2      RETURN true
: 865      1157 1      END;

```

46 49 44 2E 0022C P.ACM:

.PSECT \$SPLITS,NOWRT,NOEXE,2

.ASCII \.DIF\

.EXTRN SYSS\$CREATE

.PSECT \$CODE\$,NOWRT,2

.ENTRY DIFSOPEN_OUT, Save R2,R3,R4,R5,R6,R7,R8,R9,-; R10,R11 1097

```

OFFC 00000
0050 8F      00      5B 00000000G 00 9E 00002
      5A 00000000G 00 9E 00009
      59 00000000G 00 9E 00010
      58 00000000G 00 9E 00017
      57 00000000G 00 9E 0001E
      5E      B0      AE 9E 00025
      6E      00      2C 00029
      6E      5003    8F B0 00031
      04 AE 20000000 8F D0 00036
      16 AE      01 90 0003E
      1E AE      0202 8F B0 00042
      28 AE      67 9E 00048
      2C AE      04  A9 D0 0004C
      30 AE 00000000' EF 9E 00051
      34 AE      69 90 00059
      35 AE      04 90 0005D
0060 8F      00      6E      00      2C 00061
      67      00068
      67      6002    8F B0 00069
      02 A7      01 8E 0006E
      04 A7      6B 9E 00072
      0A A7      01 8E 00076
      0C A7      6B 9E 0007A
      10 A7 00000000G 00 D0 0007E
      5E DD 00086
      00000000G 00 01 FB 00088
      56      50 D0 0008F
      FBFC CF      5E DD 00092
      01 FB 00094
      50 DD 00099
      59 DD 0009B
      00000000G 00 02 FB 0009D
      0D      50 E8 000A4
      50 DD 000A7
      7E D4 000A9
      00000000G 8F DD 000AB
      6A      03 FB 000B1
      12      56 E8 000B4 18:
      0C AE DD 000B7

```

```

MOVAB DIFSGL_OUTRSS, R11
MOVAB LIB$STOP, R10
MOVAB DIFSGL_OUTDESC, R9
MOVAB $RMS_PTR, R8
MOVAB DIFSGL_OUTNAM, R7
MOVAB -80(SP), SP
MOVCS #0, (SP), #0, #80, $RMS_PTR 1117

MOVW #20483, $RMS_PTR
MOVL #536870912, $RMS_PTR+4
MOVB #1, $RMS_PTR+22
MOVW #514, $RMS_PTR+30
MOVAB DIFSGL_OUTNAM, $RMS_PTR+40
MOVL DIFSGL_OUTDESC+4, $RMS_PTR+44
MOVAB P.ACM, $RMS_PTR+48
MOVB DIFSGL_OUTDESC, $RMS_PTR+52
MOVB #4, $RMS_PTR+53
MOVCS #0, (SP), #0, #96, $RMS_PTR 1124

MOVW #24578, $RMS_PTR
MNEGB #1, $RMS_PTR+2
MOVAB DIFSGL_OUTRSS, $RMS_PTR+4
MNEGB #1, $RMS_PTR+10
MOVAB DIFSGL_OUTRSS, $RMS_PTR+12
MOVL DIFSGL_MASNAM, $RMS_PTR+16 1128
PUSHL SP
CALLS #1, SYSS$CREATE
MOVL R0, STATUS 1129
PUSHL SP
CALLS #1, GETFILENAME
PUSHL R0
PUSHL R9
CALLS #2, STR$COPY DX
BLBS STR$STATUS, 18 1131
PUSHL STR$STATUS 1132
CLRL -(SP)
PUSHL #DIFS_STR$CPYERR
CALLS #3, LIB$STOP
BLBS STATUS, 28 1133
PUSHL FAB+12 1135

```

| | | | | | | | | | | | |
|------|----|----|-----------|----|-----------|--------------|------|-------------|--------|------------------------------|------|
| 0044 | 8F | 00 | 00000000G | 6A | 00000000G | 00 | 02 | 56 DD 000BA | PUSHL | STATUS | |
| | | | | 6E | | 00 | | 59 DD 000BC | PUSHL | R9 | 1134 |
| | | | | | | 01 | | 01 DD 000BE | PUSHL | #1 | |
| | | | | | | 8F | | 8F DD 000C0 | PUSHL | #DIFS_OPENOUT | |
| | | | | | | 05 | | 05 FB 000C6 | CALLS | #5, LIB\$STOP | |
| | | | | | | AE | | AE 3C 000C9 | MOVZWL | FAB+2, DIFSGL_OUTIFI | 1137 |
| | | | | | | 00 | | 00 2C 000D1 | MOVCS | #0, (SP), #0, #68, \$RMS_PTR | 1144 |
| | | | | | | 68 | | 68 000D8 | | | |
| | | | | | | 68 | 4401 | 8F B0 000D9 | MOVW | #17409, \$RMS_PTR | |
| | | | | | | | 1E | AB 94 000DE | CLRB | \$RMS_PTR+30 | |
| | | | | | | 3C | | 6E 9E 000E1 | MOVAB | FAB, -\$RMS_PTR+60 | |
| | | | | | | 00000000G | | 58 DD 000E5 | PUSHL | R8 | 1146 |
| | | | | | | 00 | | 01 FB 000E7 | CALLS | #1, SYSSCONNECT | |
| | | | | | | 56 | | 50 DD 000EE | MOVL | R0, STATUS | |
| | | | | | | 12 | | 56 EB 000F1 | BLBS | STATUS, 3\$ | |
| | | | | | | | 0C | AB DD 000F4 | PUSHL | DIFSGL_OUTRAB+12 | 1148 |
| | | | | | | | | 56 DD 000F7 | PUSHL | STATUS | |
| | | | | | | | | 59 DD 000F9 | PUSHL | R9 | 1147 |
| | | | | | | 01 | | 01 DD 000FB | PUSHL | #1 | |
| | | | | | | 00000000G | | 8F DD 000FD | PUSHL | #DIFS_OPENOUT | |
| | | | | | | 6A | | 05 FB 00103 | CALLS | #5, LIB\$STOP | |
| | | | | | | 09 00000000G | 00 | 01 E0 00106 | BBS | #1, DIFSGL_FLAGS+1, 4\$ | 1153 |
| | | | | | | | | 5E DD 0010E | PUSHL | SP | 1154 |
| | | | | | | 00000000V | EF | 01 FB 00110 | CALLS | #1, FIND_LIST_WIDTH | |
| | | | | | | | 50 | 01 DD 00117 | MOVL | #1, R0 | 1156 |
| | | | | | | | | 04 0011A | RET | | 1157 |

; Routine Size: 283 bytes, Routine Base: \$CODE\$ + 0937

```

867 1158 1 GLOBAL ROUTINE dif$close_in (fdb) =
868 1159 2 BEGIN
869 1160 2
870 1161 2 Close an input file
871 1162 2
872 1163 2 Inputs:
873 1164 2
874 1165 2 fdb address of associated FDB
875 1166 2
876 1167 2 MAP
877 1168 2 fdb : REF BBLOCK;
878 1169 2
879 1170 2 BIND
880 1171 2 inrab = .fdb[fdb$l_rabptr] : BBLOCK;
881 1172 2
882 1173 2 LOCAL
883 1174 2 status,
884 1175 2 fab : $FAB_DECL;
885 1176 2
886 1177 2 $FAB_INIT(fab=fab);
887 1178 2 fab[fab$l_ifi] = .fdb[fdb$l_ifi];
888 1179 2
889 1180 2 Disconnect the record stream
890 1181 2
891 1182 2 inrab[rab$l_fab] = fab;
892 1183 2 status = $DISCONNECT(rab=inrab);
893 1184 2 IF NOT .status
894 1185 2 THEN SIGNAL(dif$closein,1,.fdb[fdb$l_fildesc],
895 1186 2 .status,.inrab[rab$l_stv]);
896 1187 2
897 1188 2 status = $CLOSE(fab=fab);
898 1189 2 IF NOT .status
899 1190 2 THEN SIGNAL(dif$closein,1,.fdb[fdb$l_fildesc],
900 1191 2 .status,.fab[fab$l_stv]);
901 1192 2
902 1193 2 RETURN true
903 1194 1 END;

```

.EXTRN SYSSDISCONNECT, SYSSCLOSE

| | | | | | | |
|------|----|-----------|-----------|-------------|--|--------|
| | | | | 03FC 0000 | .ENTRY DIF\$CLOSE IN, Save R2,R3,R4,R5,R6,R7,R8,R9 | : 1158 |
| | | 59 | 00000000G | 00 9E 00002 | MOVAB LIB\$SIGNAC, R9 | |
| | | 58 | 00000000G | 8F D0 00009 | MOVL #DIF\$CLOSEIN, R8 | |
| | | 5E | B0 | AE 9E 00010 | MOVAB -80(SP), SP | |
| | | 57 | 04 | AC D0 00014 | MOVL FDB, R7 | : 1171 |
| | | 56 | 30 | A7 D0 00018 | MOVL 48(R7), R6 | |
| 0050 | 8F | 00 | | 00 2C 0001C | MOVCS #0, (SP), #0, #80, \$RMS_PTR | : 1177 |
| | | | | 6E 00023 | | |
| | | 6E | 5003 | 8F B0 00024 | MOVW #20483, \$RMS_PTR | |
| | | 16 | | 02 90 00029 | MOVW #2, \$RMS_PTR+22 | |
| | | 1F | | 02 90 0002D | MOVW #2, \$RMS_PTR+31 | |
| | | 02 | | A7 B0 00031 | MOVW 44(R7), FAB+2 | : 1178 |
| | | 3C | A6 | 6E 9E 00036 | MOVAB FAB, 60(R6) | : 1182 |
| | | | | 56 DD 0003A | PUSHL R6 | : 1183 |
| | | 00000000G | 00 | 01 FB 0003C | CALLS #1, SYSSDISCONNECT | |

DIF_GETCMD
V04=000

DIF\$GETCMD Process command line

M 15
15-Sep-1984 23:39:15
14-Sep-1984 12:19:20

VAX-11 Bliss-32 V4.0-742
[DIF.SRC]DIFGETCMD.B32;1

Page 43
(13)

| | | | | | | | | |
|-----------|----|----|----|-------|-------|-----------------|--|------|
| 52 | | 50 | DO | 00043 | MOVL | R0, STATUS | | |
| 0F | | 52 | EB | 00046 | BLBS | STATUS, 1\$ | | 1184 |
| | 0C | A6 | DD | 00049 | PUSHL | 12(R6) | | 1186 |
| | | 52 | DD | 0004C | PUSHL | STATUS | | |
| | 38 | A7 | DD | 0004E | PUSHL | 56(R7) | | 1185 |
| | | 01 | DD | 00051 | PUSHL | #1 | | |
| | | 58 | DD | 00053 | PUSHL | R8 | | |
| 69 | | 05 | FB | 00055 | CALLS | #5, LIB\$SIGNAL | | |
| | | 5E | DD | 00058 | PUSHL | SP | | 1188 |
| 00000000G | 00 | 01 | FB | 0005A | CALLS | #1, SYSS\$CLOSE | | |
| | 52 | 50 | DO | 00061 | MOVL | R0, STATUS | | |
| | 0F | 52 | EB | 00064 | BLBS | STATUS, 2\$ | | 1189 |
| | | 0C | AE | 00067 | PUSHL | FAB+12 | | 1191 |
| | | 52 | DD | 0006A | PUSHL | STATUS | | |
| | 38 | A7 | DD | 0006C | PUSHL | 56(R7) | | 1190 |
| | | 01 | DD | 0006F | PUSHL | #1 | | |
| | | 58 | DD | 00071 | PUSHL | R8 | | |
| 69 | | 05 | FB | 00073 | CALLS | #5, LIB\$SIGNAL | | |
| | 50 | 01 | DO | 00076 | MOVL | #1, R0 | | 1193 |
| | | 04 | DD | 00079 | RET | | | 1194 |

; Routine Size: 122 bytes, Routine Base: \$CODE\$ + 0A52

DIF\$GETCMD Process command line

```

: 905      1195 1 GLOBAL ROUTINE dif$close_out =
: 906      1196 2 BEGIN
: 907      1197 2
: 908      1198 2   Close the output file
: 909      1199 2
: 910      1200 2 LOCAL
: 911      1201 2   status
: 912      1202 2   fab : $FAB_DECL;
: 913      1203 2
: 914      1204 2 $FAB_INIT(fab=fab);
: 915      1205 2 fab[fab$w_ifi] = .dif$gl_outifi;
: 916      1206 2
: 917      1207 2   Disconnect the record stream
: 918      1208 2
: 919      1209 2 status = $DISCONNECT(rab=dif$gl_outrab);
: 920      1210 2 IF NOT .status
: 921      1211 2   THEN SIGNAL(dif$closeout,1,dif$gl_outdesc,
: 922      1212 2   .status,.dif$gl_outrab[rab$l_stv]);
: 923      1213 2
: 924      1214 2 status = $CLOSE(fab=fab);
: 925      1215 2 IF NOT .status
: 926      1216 2   THEN SIGNAL(dif$closeout,1,dif$gl_outdesc,
: 927      1217 2   .status,.fab[fab$l_stv]);
: 928      1218 2
: 929      1219 2 RETURN true
: 930      1220 1 END;

```

| | | | | | | | |
|------|----|----|-----------|-------------|--------|---|--------|
| | | | | 01FC 00000 | .ENTRY | DIF\$CLOSE_OUT, Save R2,R3,R4,R5,R6,R7,R8 | : 1195 |
| | | 58 | 00000000G | 00 9E 00002 | MOVAB | LIB\$SIGNAL, R8 | |
| | | 57 | 00000000G | 8F D0 00009 | MOVL | #DIF\$CLOSEOUT, R7 | |
| | | 56 | 00000000G | 00 9E 00010 | MOVAB | DIF\$GL_OUTDESC, R6 | |
| | | 5E | B0 | AE 9E 00017 | MOVAB | -80(SPT, SP | |
| 0050 | 8F | | | 00 2C 0001B | MOVCS | #0, (SP), #0, #80, \$RMS_PTR | : 1204 |
| | | | | 6E | | | |
| | | 6E | 5003 | 8F B0 00023 | MOVW | #20483, \$RMS_PTR | |
| | 16 | AE | | 02 90 00028 | MOVW | #2, \$RMS_PTR+22 | |
| | 1F | AE | | 02 90 0002C | MOVW | #2, \$RMS_PTR+31 | |
| | 02 | AE | 00000000G | 00 B0 00030 | MOVW | DIF\$GL_OUTIFI, FAB+2 | : 1205 |
| | | | 00000000G | 00 9F 00038 | PUSHAB | DIF\$GL_OUTRAB | : 1209 |
| | | 00 | | 01 FB 0003E | CALLS | #1, SYS\$DISCONNECT | |
| | | 52 | | 50 D0 00045 | MOVL | R0, STATUS | |
| | | 11 | | 52 E8 00048 | BLBS | STATUS, 1\$ | : 1210 |
| | | | 00000000G | 00 DD 0004B | PUSHL | DIF\$GL_OUTRAB+12 | : 1212 |
| | | | | 52 DD 00051 | PUSHL | STATUS | |
| | | | | 56 DD 00053 | PUSHL | R6 | : 1211 |
| | | | | 01 DD 00055 | PUSHL | #1 | |
| | | | | 57 DD 00057 | PUSHL | R7 | |
| | | 68 | | 05 FB 00059 | CALLS | #5, LIB\$SIGNAL | |
| | | | | 5E DD 0005C | PUSHL | SP | : 1214 |
| | | 00 | 00000000G | 01 FB 0005E | CALLS | #1, SYS\$CLOSE | |
| | | 52 | | 50 D0 00065 | MOVL | R0, STATUS | |
| | | 0E | | 52 E8 00068 | BLBS | STATUS, 2\$ | : 1215 |
| | | | 0C | AE DD 0006B | PUSHL | FAB+12 | : 1217 |

DIF_GETCMD
V04=000

DIFSGETCMD Process command line

B 16
15-Sep-1984 23:39:15
14-Sep-1984 12:19:20

VAX-11 Bliss-32 V4.0-742
[DIF.SRC]DIFGETCMD.B32;1

Page 45
(14)

| | | | | | | |
|----|----|----|-------|-------|----------------|--------|
| | 52 | DD | 0006E | PUSHL | STATUS | : |
| | 56 | DD | 00070 | PUSHL | R6 | : 1216 |
| | 01 | DD | 00072 | PUSHL | #1 | : |
| | 57 | DD | 00074 | PUSHL | R7 | : |
| 68 | 05 | FB | 00076 | CALLS | #5, LIBSSIGNAL | : |
| 50 | 01 | DD | 00079 | MOVL | #1, R0 | : 1219 |
| | 04 | DD | 0007C | RET | | : 1220 |

; Routine Size: 125 bytes. Routine Base: \$CODE\$ + 0ACC

DIF\$GETCMD Process command line

```

: 932      1221 1 ROUTINE initfile(fdb,rss,fab) =
: 933      1222 2 BEGIN
: 934      1223 2
: 935      1224 2 : Initialize data structures in preparation of
: 936      1225 2 : opening a file
: 937      1226 2
: 938      1227 2 : Inputs:
: 939      1228 2
: 940      1229 2 :         fdb           Address of file's FDB
: 941      1230 2 :         rss           Address of resultant name string (optional)
: 942      1231 2 :         fab           Address of FAB
: 943      1232 2
: 944      1233 2 MAP
: 945      1234 2     fdb : REF BBLOCK,
: 946      1235 2     fab : REF BBLOCK;
: 947      1236 2
: 948      1237 2 BUILTIN
: 949      1238 2     NULLPARAMETER;
: 950      1239 2
: 951      1240 2 BIND
: 952      1241 2     rab = .fdb[fdb$_rabptr] : BBLOCK,
: 953      1242 2     nam = .fdb[fdb$_namptr] : BBLOCK,
: 954      1243 2     fildesc = .fdb[fdb$_fildesc] : BBLOCK;
: 955      1244 2
: 956      1245 2 :
: 957      1246 2 : Initialize the FAB
: 958      1247 2
: 959      1248 2 IF NOT NULLPARAMETER(2)
: 960      1249 2 THEN BEGIN
: 961      1250 2     $FAB_INIT(fab=.fab,
: 962      1251 2     nam=nam,
: 963      1252 2     fns=.fildesc[dsc$_length],
: 964      1253 2     fna=.fildesc[dsc$_pointer],
: 965      1254 2     fac=get);
: 966      1255 2
: 967      1256 2 :
: 968      1257 2 : Initialize the NAM block
: 969      1258 2
: 970      1259 2     $NAM_INIT(nam=nam,
: 971      1260 2     esa=.rss,
: 972      1261 2     ess=nam$_maxrss,
: 973      1262 2     rsa=.rss,
: 974      1263 2     rss=nam$_maxrss);
: 975      1264 2     END;
: 976      1265 2
: 977      1266 2 :
: 978      1267 2 : If no possibility of need to access input records by RFA,
: 979      1268 2 : then optimize file I/O.
: 980      1269 2
: 981      1270 2 IF NOT .dif$_gl ignore [ign$_v exact]
: 982      1271 2 OR ((.dif$_gl ignore AND NOT (ign$_m_exact OR igr$_m_pretty)) EQL 0)
: 983      1272 2 THEN fab [fab$_v_sqo] = true;
: 984      1273 2
: 985      1274 2 :
: 986      1275 2 : Initialize the RAB
: 987      1276 2
: 988      1277 2 $RAB_INIT(rab=rab,

```



```

: 995      1283 1 ROUTINE allocate_ubf(xab,rab) =
: 996      1284 2 BEGIN
: 997      1285 3
: 998      1286 4 Allocate a user buffer for records that cross block
: 999      1287 5 boundaries.
1000      1288 6
1001      1289 7 Inputs:
1002      1290 8
1003      1291 9         xab           Address of FHC xab
1004      1292 10        rab           Address of RAB
1005      1293 11
1006      1294 12 MAP
1007      1295 13     xab : REF BBLOCK,
1008      1296 14     rab : REF BBLOCK;
1009      1297 15
1010      1298 16 LOCAL
1011      1299 17     blocksize,
1012      1300 18     status;
1013      1301 19
1014      1302 20     blocksize = MAXU(MAXU(.xab[xab$w_lrl],.xab[xab$w_mrz]),512);
1015      1303 21
1016      1304 22 IF NOT (status = lib$get_vm(blocksize,rab[rab$l_ubf]))
1017      1305 23     THEN SIGNAL_STOP(.status);          !***
1018      1306 24
1019      1307 25     rab[rab$w_usz] = .blocksize;
1020      1308 26
1021      1309 27 RETURN true
: 1022      1310 28 END;

```

```

0004 00000 ALLOCATE_UBF:
: 1283
: 1302
: 1304
: 1305
: 1307
: 1309
: 1310

```

| | | | | | | | |
|-----------|------|----|-------|-------|--------|-------------------|------|
| 5E | | 04 | C2 | 00002 | WORD | Save R2 | |
| 50 | 04 | AC | D0 | 00005 | SUBL2 | #4, SP | |
| 51 | 0A | A0 | 3C | 00009 | MOVL | XAB, R0 | |
| 51 | 18 | A0 | B1 | 0000D | MOVZWL | 10(R0), R1 | |
| | | 04 | 1B | 00011 | CMPW | 24(R0), R1 | |
| 51 | 18 | A0 | 3C | 00013 | BLEQU | 1\$ | |
| 0200 | 8F | 51 | B1 | 00017 | MOVZWL | 24(R0), R1 | |
| | | 05 | 1E | 0001C | CMPW | R1, #512 | |
| 51 | 0200 | 8F | 3C | 0001E | BGEQU | 2\$ | |
| 6E | | 51 | D0 | 00023 | MOVZWL | #512, R1 | |
| 52 | 08 | AC | D0 | 00026 | MOVL | R1, BLOCKSIZE | |
| | 24 | A2 | 9F | 0002A | MOVL | RAB, R2 | |
| | 04 | AE | 9F | 0002D | PUSHAB | 36(R2) | 1304 |
| 0G000000G | 00 | 02 | FB | 00030 | PUSHAB | BLOCKSIZE | |
| | 09 | 50 | E8 | 00037 | CALLS | #2, LIB\$GET_VM | |
| | | 50 | DD | 0003A | BLBS | STATUS, 3\$ | |
| 00000000G | 00 | 01 | FB | 0003C | PUSHL | STATUS | 1305 |
| | 20 | 6E | B0 | 00043 | CALLS | #1, LIB\$STOP | |
| | 50 | 01 | D0 | 00047 | MOVW | BLOCKSIZE, 32(R2) | 1307 |
| | | 04 | 0004A | | MOVL | #1, R0 | 1309 |
| | | | | | RET | | 1310 |

: Routine Size: 75 bytes. Routine Base: \$CODE\$ + 0BE5

DIF_GETCMD
V04=000

DIFSGETCMD Process command line

F 16
15-Sep-1984 23:39:15
14-Sep-1984 12:19:20

VAX-'' Bliss-32 V4.0-742
[DIF.SRC]DIFGETCMD.B32;1

Page 49
(16)

```

: 1024 1311 1 ROUTINE find_list_width (fab) =
: 1025 1312 2 BEGIN
: 1026 1313 2
: 1027 1314 2 : Determine the width of the listing line
: 1028 1315 2 : FAB is the fab of the open file, width returned in dif$gl_width
: 1029 1316 2
: 1030 1317 2 MAP
: 1031 1318 2   fab : REF BBLOCK;
: 1032 1319 2
: 1033 1320 2 BIND
: 1034 1321 2   namblk = .fab [fab$l_nam] : BBLOCK;           !NAM block
: 1035 1322 2
: 1036 1323 2 LOCAL
: 1037 1324 2   status,
: 1038 1325 2   devnamdesc : BBLOCK [dsc$c_s_bln],
: 1039 1326 2   devnambuf : VECTOR [nam$c_maxrss, BYTE],
: 1040 1327 2   devnambufdesc : BBLOCK [dsc$c_s_bln],
: 1041 1328 2   devinfobuf : BBLOCK [dib$k_length],
: 1042 1329 2   devinfodesc : BBLOCK [dsc$c_s_bln],
: 1043 1330 2   devchan;
: 1044 1331 2
: 1045 1332 2 LITERAL
: 1046 1333 2   ch_escape = %X '1B';           !ASCII <ESC>
: 1047 1334 2
: 1048 1335 2 dif$gl_width = dif$c_deflisiz;           !Assume default of 80
: 1049 1336 2
: 1050 1337 2 devnamdesc[dsc$a_pointer] = .namblk[nam$l_dev];
: 1051 1338 2 devnamdesc[dsc$w_length] = CH$FIND_CH(.namblk[nam$b_dev],
: 1052 1339 2   .namblk[nam$l_dev], '%ASCII ':')
: 1053 1340 2   - .namblk[nam$l_dev];
: 1054 1341 2 devnambufdesc [dsc$w_length] = nam$c_maxrss;
: 1055 1342 2 devnambufdesc [dsc$a_pointer] = devnambuf;
: 1056 1343 2 $STRNLOG (LOGNAM = devnamdesc, RSLLEN = devnambufdesc, RSLBUF = devnambufdesc);
: 1057 1344 2 IF .devnambuf [0] EQL ch_escape           !Check for process permanent file
: 1058 1345 2 THEN BEGIN
: 1059 1346 2   devnambufdesc [dsc$w_length] = .devnambufdesc [dsc$w_length] - 4;
: 1060 1347 2   devnambufdesc [dsc$a_pointer] = .devnambufdesc [dsc$a_pointer] + 4;
: 1061 1348 2   END;
: 1062 1349 2
: 1063 1350 2 : Assign the device and then do a $GETCHN to get the width
: 1064 1351 2
: 1065 1352 2 IF $ASSIGN (DEVNAM = devnambufdesc, CHAN = devchan)
: 1066 1353 2 THEN BEGIN
: 1067 1354 2   devinfodesc [dsc$w_length] = dib$k_length;           !Set up descriptor for $GETCHN
: 1068 1355 2   devinfodesc [dsc$a_pointer] = devinfobuf;
: 1069 1356 2   IF $GETCHN (CHAN = .devchan, SCDBUF = devinfodesc)
: 1070 1357 2     THEN dif$gl_width = MINU(.devinfobuf [dib$w_devbufsiz], dif$c_maxlisiz);
: 1071 1358 2   $DASSGN (CHAN = .devchan);           !Deassign channel
: 1072 1359 2   END;
: 1073 1360 2
: 1074 1361 2 RETURN true;
: 1075 1362 1 END;           !Of find_list_width

```

```

.EXTRN  SYS$STRNLOG, SYS$ASSIGN
.EXTRN  SYS$GETCHN, SYS$DASSGN

```

| | | 000C 00000 FIND_LIST WIDTH: | | | | |
|----|-----------|-----------------------------|-----------|----|-------------|---|
| | | 53 | 00000000G | 00 | 9E 00002 | WORD Save R2,R3 : 1311 |
| | | 5E | FE70 | CE | 9E 00009 | MOVAB DIF\$GL WIDTH, R3 |
| | | 50 | 04 | AC | D0 0000E | MOVAB -400(SP), SP |
| | | 52 | 28 | A0 | D0 00012 | MOVL FAB, R0 : 1321 |
| | | 63 | 50 | 8F | 9A 00016 | MOVL 40(R0), R2 |
| | FC | AD | 44 | A2 | D0 0001A | MOVZBL #80, DIF\$GL WIDTH : 1335 |
| | | 50 | 39 | A2 | 9A 0001F | MOVL 68(R2), DEVNAMDESC+4 : 1337 |
| 44 | B2 | 50 | | 3A | 3A 00023 | MOVZBL 57(R2), R0 : 1338 |
| | | | | 02 | 12 00028 | LOCC #58, R0, @68(R2) |
| | | | | 51 | D4 0002A | BNEQ 1\$ |
| F8 | AD | 51 | 44 | A2 | A3 0002C | CLRL R1 : 1340 |
| | 0080 | CE | FF | 8F | 9B 00032 | SUBW3 68(R2), R1, DEVNAMDESC : 1341 |
| | 0084 | CE | 0088 | CE | 9E 00038 | MOVZBW #255, DEVNAMBUFDESC : 1342 |
| | | | | 7E | 7C 0003F | MOVAB DEVNAMBUF, DEVNAMBUFDESC+4 : 1343 |
| | | | | 7E | D4 00041 | CLRL -(SP) |
| | | | 008C | CE | 9F 00043 | CLRL -(SP) |
| | | | 0090 | CE | 9F 00047 | PUSHAB DEVNAMBUFDESC |
| | | | F8 | AD | 9F 0004B | PUSHAB DEVNAMBUFDESC |
| | 00000000G | 00 | | 06 | FB 0004E | PUSHAB DEVNAMDESC |
| | | 1B | 0088 | CE | 91 00055 | CALLS #6, SYS\$STRNLOG : 1344 |
| | | | | 0A | 12 0005A | CMPB DEVNAMBUF, #27 |
| | 0080 | CE | | 04 | A2 0005C | BNEQ 2\$ |
| | 0084 | CE | | 04 | C0 00061 | SUBW2 #4, DEVNAMBUFDESC : 1346 |
| | | | | 7E | 7C 00066 | ADDL2 #4, DEVNAMBUFDESC+4 : 1347 |
| | | | | 08 | AE 9F 00068 | CLRL -(SP) : 1352 |
| | | | 008C | CE | 9F 0006B | PUSHAB DEVCHAN |
| | 00000000G | 00 | | 04 | FB 0006F | PUSHAB DEVNAMBUFDESC |
| | | 39 | | 50 | E9 00076 | CALLS #4, SYSSASSIGN |
| | 04 | AE | 74 | 8F | 9B 00079 | BLBC R0, 5\$ |
| | 08 | AE | 0C | AE | 9E 0007E | MOVZBW #116, DEVINFODESC : 1354 |
| | | | 04 | AE | 9F 00083 | MOVAB DEVINFOBUF, DEVINFODESC+4 : 1355 |
| | | | | 7E | 7C 00086 | PUSHAB DEVINFODESC : 1356 |
| | | | | 7E | D4 00088 | CLRL -(SP) |
| | | | 10 | AE | DD 0008A | CLRL -(SP) |
| | 00000000G | 00 | | 05 | FB 0008D | PUSHL DEVCHAN |
| | | 12 | | 50 | E9 00094 | CALLS #5, SYSSGETCHN |
| | | 50 | 12 | AE | 3C 00097 | BLBC R0, 4\$ |
| | 0084 | 8F | | 50 | B1 0009B | MOVZWL DEVINFOBUF+6, R0 : 1357 |
| | | | | 04 | 1B 000A0 | CMPW R0, #132 |
| | | 50 | 84 | 8F | 9A 000A2 | BLEQU 3\$ |
| | | 63 | | 50 | D0 000A6 | MOVZBL #132, R0 |
| | | | | 6E | DD 000A9 | MOVL R0, DIF\$GL_WIDTH : 1358 |
| | 00000000G | 00 | | 01 | FB 000AB | PUSHL DEVCHAN |
| | | 50 | | 01 | D0 000B2 | CALLS #1, SYSSDASSGN : 1361 |
| | | | | 04 | 000B5 | MOVL #1, R0 : 1362 |
| | | | | | | RET |

; Routine Size: 182 bytes, Routine Base: \$CODE\$ + 0C30

; 1076 1363 1 END
; 1077 1364 0 ELUDOM

' Of module DIFGETCMD

.EXTRN LIB\$SIGNAL, LIB\$STOP

PSECT SUMMARY

| Name | Bytes | Attributes |
|--------------|-------|--|
| \$OWNS | 76 | NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2) |
| \$PLITS | 560 | NOVEC, NOWRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2) |
| _LIB\$KEYOS | 24 | NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(1) |
| _LIB\$STATES | 144 | NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(1) |
| _LIB\$KEY1S | 90 | NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(1) |
| \$CODES | 3302 | 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]STARLFT.L32;1 | 9776 | 131 | 1 | 581 | 00:01.0 |
| \$_\$255\$DUA28:[SYSLIB]TPAMAC.L32;1 | 42 | 23 | 54 | 14 | 00:00.1 |

COMMAND QUALIFIERS

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

Size: 3302 code + 894 data bytes
 Run Time: 01.04.4
 Elapsed Time: 02:40.5
 Lines/CPU Min: 1270
 Lexemes/CPU-Min: 37687
 Memory Used: 398 pages
 Compilation Complete

The image displays a grid of 100 small terminal window screenshots, arranged in 10 rows and 10 columns. Each window shows a different command or system output. The text is small and difficult to read, but several larger, legible labels are visible within the grid:

- DIFDEF MDL
- DIF
- DIF MAP
- XDSTRING LIS
- DIFPRE REQ
- DIFGETCMD LIS
- DIFHEXOCT LIS
- DATA LIS
- XDELTA LIS