```
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTTT  LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBB         BBB   AAA         AAA   SSS          RRR         RRR         TTT        LLL
BBBBBBBBBBBB      AAA         AAA     SSSSSSSSS  RRRRRRRRRRR             TTT        LLL
BBBBBBBBBBBB      AAA         AAA     SSSSSSSSS  RRRRRRRRRRR             TTT        LLL
BBBBBBBBBBBB      AAA         AAA     SSSSSSSSS  RRRRRRRRRRR             TTT        LLL
BBB         BBB   AAAAAAAAAAAAAAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAAAAAAAAAAAAAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAAAAAAAAAAAAAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAA         AAA           SSS  RRR      RRR            TTT        LLL
BBB         BBB   AAA         AAA           SSS  RRR   RRR               TTT        LLL
BBB         BBB   AAA         AAA           SSS  RRR      RRR            TTT        LLL
BBBBBBBBBBBB      AAA         AAA   SSSSSSSSSSS     RRR         RRR      TTT        LLLLLLLLLLLLLLL
BBBBBBBBBBBB      AAA         AAA   SSSSSSSSSSS     RRR         RRR      TTT        LLLLLLLLLLLLLLL
BBBBBBBBBBBB      AAA         AAA   SSSSSSSSSSS     RRR         RRR      TTT        LLLLLLLLLLLLLLL
```

```
BBBBBBBB      AAAAAA      SSSSSSSS  FFFFFFFFFF  RRRRRRRR    EEEEEEEEEE  EEEEEEEEEE
BBBBBBBB      AAAAAA      SSSSSSSS  FFFFFFFFFF  RRRRRRRR    EEEEEEEEEE  EEEEEEEEEE
BB      BB  AA      AA  SS          FF          RR      RR  EE          EE
BB      BB  AA      AA  SS          FF          RR      RR  EE          EE
BB      BB  AA      AA  SS          FF          RR      RR  EE          EE
BB      BB  AA      AA  SS          FF          RR      RR  EE          EE
BBBBBBBB    AA      AA    SSSSSS    FFFFFFFF    RRRRRRRR    EEEEEEEE    EEEEEEEE
BBBBBBBB    AA      AA    SSSSSS    FFFFFFFF    RRRRRRRR    EEEEEEEE    EEEEEEEE
BB      BB  AAAAAAAAAA          SS  FF          RR  RR      EE          EE
BB      BB  AAAAAAAAAA          SS  FF          RR  RR      EE          EE
BB      BB  AA      AA          SS  FF          RR    RR    EE          EE              ....
BB      BB  AA      AA          SS  FF          RR    RR    EE          EE              ....
BBBBBBBB    AA      AA  SSSSSSSS    FF          RR      RR  EEEEEEEEEE  EEEEEEEEEE      ....
BBBBBBBB    AA      AA  SSSSSSSS    FF          RR      RR  EEEEEEEEEE  EEEEEEEEEE      ....


LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
;     1          0001  0 MODULE BAS$FREE (                                  ! Basic FREE construct
;     2          0002  0                 IDENT = '1-003'                   ! File: BASFREE.B32
;     3          0003  0                 ) =
;     4          0004  1 BEGIN
;     5          0005  1 !
;     6          0006  1 !*****************************************************************************
;     7          0007  1 !*                                                                           *
;     8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
;     9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
;    10          0010  1 !*  ALL RIGHTS RESERVED.                                                     *
;    11          0011  1 !*                                                                           *
;    12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    13          0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    16          0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    17          0017  1 !*  TRANSFERRED.                                                             *
;    18          0018  1 !*                                                                           *
;    19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    20          0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    21          0021  1 !*  CORPORATION.                                                             *
;    22          0022  1 !*                                                                           *
;    23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
;    25          0025  1 !*                                                                           *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*****************************************************************************
;    28          0028  1 !
;    29          0029  1
;    30          0030  1 !++
;    31          0031  1 ! FACILITY:
;    32          0032  1 !       Basic support library - user callable
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !       This module is the UPI level of the Basic FREE construct.
;    36          0036  1 !       This module will setup the I/O data base for the LUN and go directly to
;    37          0037  1 !       the REC level.
;    38          0038  1 !
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:
;    41          0041  1 !       User access mode - AST reentrant.
;    42          0042  1 !
;    43          0043  1 ! AUTHOR: Donald G. Petersen, CREATION DATE: 28-Feb-79
;    44          0044  1 !
;    45          0045  1 ! MODIFIED BY:
;    46          0046  1 !
;    47          0047  1 !       DGP, 28-Feb-79 : VERSION 01
;    48          0048  1 ! 1-001 - original.  DGP 28-Feb-79
;    49          0049  1 ! 1-002 - Change the name of the REC level to FEE due to conflict with
;    50          0050  1 !         FIND relative.  DGP 02-Mar-79
;    51          0051  1 ! 1-003 - Set up ISB$A_USER_FP.  JBS 25-JUL-1979
;    52          0052  1 !--
;    53          0053  1
;    54          0054  1 !<BLF/PAGE>
```

```
;    56          0055  1 !
;    57          0056  1 ! SWITCHES:
;    58          0057  1 !
;    59          0058  1
;    60          0059  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    61          0060  1
;    62          0061  1 !
;    63          0062  1 ! LINKAGES
;    64          0063  1 !
;    65          0064  1
;    66          0065  1 REQUIRE 'RTLIN:OTSLNK';                         ! Define all linkages
;    67          0494  1
;    68          0495  1 !
;    69          0496  1 ! TABLE OF CONTENTS:
;    70          0497  1 !
;    71          0498  1
;    72          0499  1 FORWARD ROUTINE
;    73          0500  1     BAS$FREE : NOVALUE;                         ! UPI level Sequential FREE
;    74          0501  1
;    75          0502  1 !
;    76          0503  1 ! INCLUDE FILES:
;    77          0504  1 !
;    78          0505  1
;    79          0506  1 REQUIRE 'RTLML:OTSISB';                         ! ISB definitions
;    80          0674  1
;    81          0675  1 REQUIRE 'RTLML:OTSLUB';                         ! LUB definitions
;    82          0815  1
;    83          0816  1 REQUIRE 'RTLIN:RTLPSECT';                       ! Define DECLARE_PSECTS macro
;    84          0911  1
;    85          0912  1 LIBRARY 'RTLSTARLE';                            ! Starlet system macros
;    86          0913  1
;    87          0914  1 !
;    88          0915  1 ! MACROS:
;    89          0916  1 !
;    90          0917  1 !       NONE
;    91          0918  1 !
;    92          0919  1 ! EQUATED SYMBOLS:
;    93          0920  1 !
;    94          0921  1 !       NONE
;    95          0922  1 !
;    96          0923  1 !
;    97          0924  1 ! PSECT DECLARATIONS:
;    98          0925  1 !
;    99          0926  1 DECLARE_PSECTS (BAS);
;   100          0927  1 !
;   101          0928  1 ! OWN STORAGE:
;   102          0929  1 !
;   103          0930  1 !       NONE
;   104          0931  1 !
;   105          0932  1 ! EXTERNAL REFERENCES:
;   106          0933  1 !
;   107          0934  1
;   108          0935  1 EXTERNAL ROUTINE
;   109          0936  1     BAS$$STOP_IO : NOVALUE,                     ! Signal fatal BASIC I/O error
;   110          0937  1     BAS$$REC_FEE : JSB_REC0 NOVALUE,            ! REC level processing - RMS interface
;   111          0938  1                                                 ! FREE
;   112          0939  1     BAS$$CB_PUSH : JSB_CB_PUSH NOVALUE,         ! Load register CCB
```

```
;  113          0940  1     BAS$$CB_POP : JSB_CB_POP NOVALUE;           ! Done with register CCB
;  114          0941  1
;  115          0942  1 !+
;  116          0943  1 ! The following are the error codes used in this module.
;  117          0944  1 !-
;  118          0945  1
;  119          0946  1 EXTERNAL LITERAL
;  120          0947  1     BAS$K_IO_CHANOT : UNSIGNED (8);             ! I/O channel not open
;  121          0948  1
```

```
;  123     0949  1 GLOBAL ROUTINE BAS$FREE (                      ! FREE sequential
;  124     0950  1         UNIT                                   ! logical unit number
;  125     0951  1     ) : NOVALUE =
;  126     0952  1
;  127     0953  1 !++
;  128     0954  1 ! FUNCTIONAL DESCRIPTION:
;  129     0955  1 !
;  130     0956  1 !       This routine will set up the I/O data base for this LUN if necessary
;  131     0957  1 !       and then go directly to the REC level.  When control is returned to
;  132     0958  1 !       this routine, it pops the CCB off of the I/O system.  The actual inter-
;  133     0959  1 !       face to RMS is done at the REC level.  All locked records are unlocked.
;  134     0960  1 !
;  135     0961  1 ! FORMAL PARAMETERS:
;  136     0962  1 !
;  137     0963  1 !       UNIT.rlu.v      logical unit number
;  138     0964  1 !
;  139     0965  1 ! IMPLICIT INPUTS:
;  140     0966  1 !
;  141     0967  1 !       NONE
;  142     0968  1 !
;  143     0969  1 ! IMPLICIT OUTPUTS:
;  144     0970  1 !
;  145     0971  1 !       ISB$B_STTM_TYPE         the statement type
;  146     0972  1 !
;  147     0973  1 ! COMPLETION CODES:
;  148     0974  1 !
;  149     0975  1 !       NONE
;  150     0976  1 !
;  151     0977  1 ! SIDE EFFECTS:
;  152     0978  1 !
;  153     0979  1 !       NONE
;  154     0980  1 !
;  155     0981  1 !--
;  156     0982  1
;  157     0983  2     BEGIN
;  158     0984  2
;  159     0985  2     BUILTIN
;  160     0986  2         FP;
;  161     0987  2
;  162     0988  2     GLOBAL REGISTER
;  163     0989  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  164     0990  2
;  165     0991  2     LOCAL
;  166     0992  2         FMP : REF BLOCK [, BYTE];
;  167     0993  2
;  168     0994  2     FMP = .FP;
;  169     0995  2 !+
;  170     0996  2 ! Allocate the LUB/ISB/RAB for this unit if necessary.  Store new CB (con-
;  171     0997  2 ! trol block) in OTS$$A_CUR_LUB.  Store signed unit number in LUB$W_LUN.
;  172     0998  2 !-
;  173     0999  2     BAS$$CB_PUSH (.UNIT, LUB$K_ILUN_MIN);
;  174     1000  2     CCB [ISB$A_USER_FP] = .FMP [SF$L_SAVE_FP];
;  175     1001  2 !+
;  176     1002  2 ! If the channel is not open, give an error message.
;  177     1003  2 ! FREE is not permitted on channel 0.
;  178     1004  2 !-
;  179     1005  2
```

```
;  180          1006  2     IF ( NOT .CCB [LUB$V_OPENED]) THEN BAS$$STOP_IO (BAS$K_IO_CHANOT);
;  181          1007  2
;  182          1008  2 !+
;  183          1009  2 ! Now that the data base is in place, store the statement type and  go
;  184          1010  2 ! directly to the REC level.
;  185          1011  2 !-
;  186          1012  2     CCB [ISB$B_STTM_TYPE] = ISB$K_ST_TY_FEE;
;  187          1013  2     BAS$$REC_FEE ();
;  188          1014  2 !+
;  189          1015  2 ! Now that the FREE has been done, pop the CCB off the I/O system.
;  190          1016  2 !-
;  191          1017  2     BAS$$CB_POP ();
;  192          1018  1     END;                                          !End of BAS$FREE

                                                                        .TITLE  BAS$FREE
                                                                        .IDENT  \1-003\

                                                                        .EXTRN  BAS$$STOP_IO, BAS$$REC_FEE
                                                                        .EXTRN  BAS$$CB_PUSH, BAS$$CB_POP
                                                                        .EXTRN  BAS$K_IO_CHANOT

                                                                        .PSECT  _BAS$CODE,NOWRT,  SHR,  PIC,2

                                    083C 00000          .ENTRY  BAS$FREE, Save R2,R3,R4,R5,R11                  ; 0949
                          53        5D   D0 00002       MOVL    FP, FMP                                         ; 0994
                          50        08   CE 00005       MNEGL   #8, R0                                          ; 0999
                          52  04    AC   D0 00008       MOVL    UNIT, R2                                        ;
                 000000000G  00    16 0000C            JSB     BAS$$CB_PUSH                                    ;
                    rF4C  CB  0C    A3   D0 00012       MOVL    12(FMP), -180(CCB)                              ; 1000
                          0B  FC    AB   E8 00018       BLBS    -4(CCB), 1$                                     ; 1006
                          7E  00G   8F   9A 0001C       MOVZBL  #BAS$K_IO_CHANOT, -(SP)                         ;
                 000000000G  00     01   FB 00020       CALLS   #1, BAS$$STOP_IO                                ;
                    FF71  CB                2B   90 00027 1$:   MOVB    #43, -143(CCB)                          ; 1012
                          000000000G  00    16 0002C            JSB     BAS$$REC_FEE                            ; 1013
                          000000000G  00    16 00032            JSB     BAS$$CB_POP                             ; 1017
                                      04 00038              RET                                               ; 1018

; Routine Size:  57 bytes,    Routine Base:  _BAS$CODE + 0000

;  193          1019  1
;  194          1020  1 END                                           ! End of module BAS$FREE
;  195          1021  1
;  196          1022  0 ELUDOM


;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  _BAS$CODE                      57  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)
```

```
;                              Library Statistics
;
;                                        --------- Symbols ---------      Pages       Processing
;       File                             Total     Loaded    Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1      9776          1          0       581          00:01.1
```

```
;                                    COMMAND QUALIFIERS

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

; Size:          57 code + 0 data bytes
; Run Time:         00:08.2
; Elapsed Time:     00:17.0
; Lines/CPU Min:     7514
; Lexemes/CPU-Min: 44683
; Memory Used:  108 pages
; Compilation Complete
```

BASFREE
LIS

BASEXITHA
LIS

BASFETCHD
LIS

BASFORINI
LIS

BASGETRFA
LIS

BASFETCHA
LIS

BASGET
LIS

BASFSP
LIS

BASFIND
LIS

BASFORMAT
LIS

BASHANDLE
LIS