```
BBBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS    RRRRRRRRRRRR    TTTTTTTTTTTTTTTT  LLL
BBBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS    RRRRRRRRRRRR    TTTTTTTTTTTTTTTTT LLL
BBBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS    RRRRRRRRRRRR    TTTTTTTTTTTTTTTTT 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
BBBBBBBBBBBBB      AAA        AAA        SSSSSSSSS    RRRRRRRRRRRR           TTT        LLL
BBBBBBBBBBBBB      AAA        AAA        SSSSSSSSS    RRRRRRRRRRRR           TTT        LLL
BBBBBBBBBBBBB      AAA        AAA        SSSSSSSSS    RRRRRRRRRRRR           TTT        LLL
BBB          BBB   AAAAAAAAAAAAAAAAA           SSS    RRR   RRR             TTT        LLL
BBB          BBB   AAAAAAAAAAAAAAAAA           SSS    RRR   RRR             TTT        LLL
BBB          BBB   AAAAAAAAAAAAAAAAA           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
BBBBBBBBBBBBB      AAA        AAA   SSSSSSSSSSSS      RRR             RRR    TTT        LLLLLLLLLLLLLL
BBBBBBBBBBBBB      AAA        AAA   SSSSSSSSSSSS      RRR             RRR    TTT        LLLLLLLLLLLLLLL
BBBBBBBBBBBBB      AAA        AAA   SSSSSSSSSSSS      RRR             RRR    TTT        LLLLLLLLLLLLLLL
```

```
BBBBBBBB     AAAAAA    SSSSSSSS   CCCCCCCC LL          000000    SSSSSSSS  EEEEEEEEEE
BBBBBBBB     AAAAAA    SSSSSSSS   CCCCCCCC LL          000000    SSSSSSSS  EEEEEEEEEE
BB     BB   AA    AA   SS         CC       LL        00    00    SS        EE
BB     BB   AA    AA   SS         CC       LL        00    00    SS        EE
BB     BB   AA    AA   SS         CC       LL        00    00    SS        EE
BBBBBBBB    AA    AA   SSSSSS     CC       LL        00    00    SSSSSS    EEEEEEEE
BBBBBBBB    AA    AA   SSSSSS     CC       LL        00    00    SSSSSS    EEEEEEEE
BB     BB   AAAAAAAAA        SS   CC       LL        00    00          SS  EE
BB     BB   AAAAAAAAA        SS   CC       LL        00    00          SS  EE
BB     BB   AA    AA         SS   CC       LL        00    00          SS  EE
BB     BB   AA    AA         SS   CC       LL        00    00          SS  EE
BBBBBBBB    AA    AA   SSSSSSSS   CCCCCCCC LLLLLLLLLL  000000    SSSSSSSS  EEEEEEEEEE   ....
BBBBBBBB    AA    AA   SSSSSSSS   CCCCCCCC LLLLLLLLLL  000000    SSSSSSSS  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
LLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLL    IIIIII    SSSSSSSS
```

```
   1   0001  0  MODULE BAS$CLOSE (                              ! Close a BASIC file
   2   0002  0                   IDENT = '1-008'                ! File: BASCLOSE.B32 EDIT:MDL1008
   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:  BASIC-PLUS-2 I/O SUPPORT
  32   0032  1  !
  33   0033  1  ! ABSTRACT:
  34   0034  1  !
  35   0035  1  !      This module contains the routine which closes a file for
  36   0036  1  !      a BASIC program.
  37   0037  1  !
  38   0038  1  ! ENVIRONMENT:  VAX-11 User Mode
  39   0039  1  !
  40   0040  1  ! AUTHOR: John Sauter, CREATION DATE: 16-FEB-1979
  41   0041  1  !
  42   0042  1  ! MODIFIED BY:
  43   0043  1  !
  44   0044  1  ! 1-001 - Original.  JBS 16-FEB-1979
  45   0045  1  ! 1-002 - Use BASIOERR.REQ to define the I/O error codes.  JBS 20-FEB-1979
  46   0046  1  ! 1-003 - Set up ISB$A_USER_FP.  JBS 25-JUL-1979
  47   0047  1  ! 1-004 - Purge the I/O buffer explicitly, since OTS$$CLOSE_FILE doesn't
  48   0048  1  !         any more.  JBS 27-AUG-1979
  49   0049  1  ! 1-005 - If the file has had more than one stream connected to it, disconnect
  50   0050  1  !         the other streams.  JBS 28-SEP-1979
  51   0051  1  ! 1-006 - Call BAS$FIELD_CLOSE to remove fielded variables from
  52   0052  1  !         the symbol table, in case the compiler doesn't. PLL 2-JUN-81
  53   0053  1  ! 1-007 - Remove the call to BAS$FIELD_CLOSE, as it is unshared and BAS$CLOSE
  54   0054  1  !         is shared.  PL 16-Jun-81
  55   0055  1  ! 1-008 - call BAS$FIELD_CLOSE now, as all BAS$ entry points are now shared.
  56   0056  1  !         MDL 26-Mar-1984
  57   0057  1  !--
```

```
;   58           0058 1
;   59           0059 1 !<BLF/PAGE>
```

```
   61   0060   1 !
   62   0061   1 ! SWITCHES:
   63   0062   1 !
   64   0063   1
   65   0064   1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
   66   0065   1
   67   0066   1 !
   68   0067   1 ! LINKAGES:
   69   0068   1 !
   70   0069   1
   71   0070   1 REQUIRE 'RTLIN:OTSLNK';                            ! Define all the linkages
   72   0499   1
   73   0500   1 !
   74   0501   1 ! TABLE OF CONTENTS:
   75   0502   1 !
   76   0503   1
   77   0504   1 FORWARD ROUTINE
   78   0505   1     BAS$CLOSE : NOVALUE;                           ! Close a file from BASIC
   79   0506   1
   80   0507   1 !
   81   0508   1 ! INCLUDE FILES:
   82   0509   1 !
   83   0510   1
   84   0511   1 REQUIRE 'RTLIN:RTLPSECT';                          ! Macros for defining psects
   85   0606   1
   86   0607   1 REQUIRE 'RTLML:OTSLUB';                            ! Define symbols in the LUB
   87   0747   1
   88   0748   1 REQUIRE 'RTLML:OTSISB';                            ! Define symbols in the ISB
   89   0916   1
   90   0917   1 REQUIRE 'RTLIN:BASIOERR';                          ! Define the I/O error codes.
   91   0970   1
   92   0971   1 LIBRARY 'RTLSTARLE';                               ! Define system symbols
   93   0972   1
   94   0973   1 !
   95   0974   1 ! MACROS:
   96   0975   1 !
   97   0976   1 !     NONE
   98   0977   1 !
   99   0978   1 ! EQUATED SYMBOLS:
  100   0979   1 !
  101   0980   1 !     NONE
  102   0981   1 !
  103   0982   1 ! PSECTS:
  104   0983   1 !
  105   0984   1 DECLARE_PSECTS (BAS);                              ! Declare psects for BAS$ facility
  106   0985   1 !
  107   0986   1 ! OWN STORAGE:
  108   0987   1 !
  109   0988   1 !     NONE
  110   0989   1 !
  111   0990   1 ! EXTERNAL REFERENCES:
  112   0991   1 !
  113   0992   1
  114   0993   1 EXTERNAL ROUTINE
  115   0994   1     BAS$$CB_PUSH : JSB_CB_PUSH NOVALUE,            ! Load register CCB
  116   0995   1     BAS$$CB_POP : JSB_CB_RET NOVALUE,              ! Done with register CCB
  117   0996   1     BAS$$STOP_IO : NOVALUE,                        ! signals fatal I/O error
```

```
;   118      0997  1      BAS$$PUR_IO_CLO : CALL_CCB,              ! Purge I/O buffer
;   119      0998  1      BAS$FIELD_CLOSE : NOVALUE,               ! delete FIELDed variables on this channel
;   120      0999  1      BAS$$CLOSE_ALL : NOVALUE,                ! Close all files with a certain IFI
;   121      1000  1      OTS$$CLOSE_FILE;                         ! RMS Close a file
;   122      1001  1
;   123      1002  1 !+
;   124      1003  1 ! The following are the error codes used in this module.
;   125      1004  1 !-
;   126      1005  1
;   127      1006  1 EXTERNAL LITERAL
;   128      1007  1      BAS$K_PROLOSSOR : UNSIGNED (8);          ! Program lost, sorry
;   129      1008  1
;   130      1009  1
```

```
132        1010  1 GLOBAL ROUTINE BASSCLOSE (                           ! Close a file
133        1011  1          CHANNEL                                     ! The BASIC channel number
134        1012  1    ) : NOVALUE =
135        1013  1
136        1014  1 !++
137        1015  1 ! FUNCTIONAL DESCRIPTION:
138        1016  1 !
139        1017  1 !     Close an I/O channel from a BASIC-PLUS-2 program.
140        1018  1 !
141        1019  1 ! FORMAL PARAMETERS:
142        1020  1 !
143        1021  1 !     CHANNEL.rl.v    The number of an open channel, which is to
144        1022  1 !                     be closed.
145        1023  1 !
146        1024  1 ! IMPLICIT INPUTS:
147        1025  1 !
148        1026  1 !     NONE
149        1027  1 !
150        1028  1 ! IMPLICIT OUTPUTS:
151        1029  1 !
152        1030  1 !     NONE
153        1031  1 !
154        1032  1 ! ROUTINE VALUE:
155        1033  1 ! COMPLETION CODES:
156        1034  1 !
157        1035  1 !     NONE
158        1036  1 !
159        1037  1 ! SIDE EFFECTS:
160        1038  1 !
161        1039  1 !     Closes the specified channel.
162        1040  1 !
163        1041  1 !--
164        1042  1
165        1043  2     BEGIN
166        1044  2
167        1045  2     BUILTIN
168        1046  2         FP;
169        1047  2
170        1048  2     GLOBAL REGISTER
171        1049  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
172        1050  2
173        1051  2     LOCAL
174        1052  2         FMP : REF BLOCK [, BYTE];
175        1053  2
176        1054  2     FMP = .FP;
177        1055  2 !+
178        1056  2 ! For compatability with the PDP-11 implementations, an attempt to
179        1057  2 ! CLOSE channel 0 is a no-operation.
180        1058  2 !-
181        1059  2
182        1060  3     IF (.CHANNEL NEQ 0)
183        1061  3     THEN
184        1062  3         BEGIN
185        1063  3
186        1064  3 !+
187        1065  3 ! Set up register CCB to point to the LUB/ISB/RAB for this channel.
188        1066  3 !-
```

```
189   1067   3              BAS$$CB_PUSH (.CHANNEL, LUB$K_LUN_MIN);
190   1068   3              CCB [ISB$A_USER_FP] = .FMP [SF$L_SAVE_FP];
191   1069   3
192   1070   3      !+
193   1071   3      ! Purge the I/O buffer.  This will only happen on a non-forcible device
194   1072   3      ! on which the last PRINT statement ended with a comma or semicolon.
195   1073   3      !-
196   1074   3              BAS$$PUR_IO_CLO ();
197   1075   3
198   1076   3      !+
199   1077   3      ! get rid of any FIELDed variables on this channel, they are no longer valid.
200   1078   3      !-
201   1079   3              IF .CCB [LUB$V_FIELD_USE]
202   1080   3              THEN
203   1081   3                  BAS$FIELD_CLOSE (.CHANNEL);
204   1082   3
205   1083   3      !+
206   1084   3      ! RMS CLOSE the file, if it is open.
207   1085   3      !-
208   1086   4              IF (.CCB [LUB$V_OPENED])
209   1087   3              THEN
210   1088   4                  BEGIN
211   1089   4      !+
212   1090   4      ! If the file might have other streams connected, disconnect them.
213   1091   4      !-
214   1092   4
215   1093   4                  IF (.CCB [LUB$V_M_STR_C]) THEN BAS$$CLOSE_ALL (.CCB [LUB$W_IFI]);
216   1094   4
217   1095   4                  IF ( NOT OTS$$CLOSE_FILE ()) THEN BAS$$STOP_IO (BAS$K_IOERR_REC);
218   1096   4
219   1097   3                  END;
220   1098   3
221   1099   3      !+
222   1100   3      ! We are done with register CCB.  If this is the only use of this channel
223   1101   3      ! (as is likely), then the LUB/ISB/RAB will be deallocated.
224   1102   3      !-
225   1103   3              BAS$$CB_POP ();
226   1104   2              END;
227   1105   2
228   1106   1          END;                                          ! end of BAS$CLOSE


                                            .TITLE   BAS$CLOSE
                                            .IDENT   \1-008\

                                            .EXTRN   BAS$$CB_PUSH, BAS$$CB_POP
                                            .EXTRN   BAS$$STOP_IO, BAS$$PUR_IO_CLO
                                            .EXTRN   BAS$FIELD_CLOSE
                                            .EXTRN   BAS$$CLOSE_ALL, OTS$$CLOSE_FILE
                                            .EXTRN   BAS$K_PROLOSSOR

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

                         080C 00000         .ENTRY   BAS$CLOSE, Save R2,R3,R11        : 1010
              53      5D   D0 00002         MOVL     FP, FMP                          : 1054
              52  04  AC   D0 00005         MOVL     CHANNEL, R2                      : 1060
              51      13   00009         BEQL     4$                                 :
```

```
                                      50  D4 0000B        CLRL    R0                                  ; 1067
                           00000000G  00  16 0000D        JSB     BAS$$CB_PUSH
                  FF4C  CB            0C  A3 D0 00013      MOVL    12(FMP), -180(CCB)                  ; 1068
             00000000G  00            00  FB 00019         CALLS   #0, BAS$$PUR_IO_CLO                 ; 1073
        09        A1  AB                  06 E1 00020      BBC     #6, -95(CCB), 1$                    ; 1078
                                      52  DD 00025         PUSHL   R2                                  ; 1080
             00000000G  00                01 FB 00027      CALLS   #1, BAS$FIELD_CLOSE
                       24            FC  AB E9 0002E 1$:   BLBC    -4(CCB), 3$                         ; 1086
        0B        FF  AB                  03 E1 00032      BBC     #3, -1(CCB), 2$                     ; 1093
                       7E            D0  AB 3C 00037       MOVZWL  -48(CCB), -(SP)
             00000000G  00                01 FB 0003B      CALLS   #1, BAS$$CLOSE_ALL
             00000000G  00            00  FB 00042 2$:     CALLS   #0, OTS$$CLOSE_FILE                 ; 1095
                       0A                 50 E8 00049      BLBS    R0, 3$
                       7E                 01 CE 0004C      MNEGL   #1, -(SP)
             00000000G  00                01 FB 0004F      CALLS   #1, BAS$$STOP_IO
                           00000000G  00  16 00056 3$:    JSB     BAS$$CB_POP                          ; 1103
                                      04  0005C 4$:       RET                                          ; 1106
```

; Routine Size: 93 bytes,   Routine Base: _BAS$CODE + 0000

```
;  229          1107  1
;  230          1108  1 END                                           ! end of module BAS$CLOSE
;  231          1109  1
;  232          1110  0 ELUDOM
```

;
;   PSECT SUMMARY
;
;   Name   Bytes   Attributes
;
;   _BAS$CODE   93 NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

;   Library Statistics
;
;   -------- Symbols --------   Pages   Processing
;   File   Total   Loaded   Percent   Mapped   Time
;
;   _255DUA28:[SYSLIB]STARLET.L32;1   9776   1   0   581   00:01.2

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

```
; Size:            93 code + 0 data bytes
; Run Time:            00:09.0
; Elapsed Time:        00:23.0
; Lines/CPU Min:     7375
; Lexemes/CPU-Min: 41534
; Memory Used:  119 pages
; Compilation Complete
```

BASCLOSE
LIS

BASCONCAT
LIS

BASCTRLO
LIS

BASCHANGE
LIS

BASCTRLC
LIS

BASCHAIN
LIS

BASCOPYFD
LIS

BASCHR
LIS

BASCMPAPP
LIS

BASCVTOUT
LIS

BASCCPOS
LIS