```
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
```

FILEID**BASCTRLO

```
BBBBBBBB      AAAAAA      SSSSSSSS   CCCCCCCC  TTTTTTTTTT  RRRRRRRR    LL          OOOOOO
BBBBBBBB      AAAAAA      SSSSSSSS   CCCCCCCC  TTTTTTTTTT  RRRRRRRR    LL          OOOOOO
BB      BB   AA      AA  SS         CC            TT      RR      RR  LL        OO      OO
BB      BB   AA      AA  SS         CC            TT      RR      RR  LL        OO      OO
BB      BB   AA      AA  SS         CC            TT      RR      RR  LL        OO      OO
BB      BB   AA      AA  SS         CC            TT      RR      RR  LL        OO      OO
BBBBBBBB     AA      AA    SSSSSS   CC            TT      RRRRRRRR    LL        OO      OO
BBBBBBBB     AA      AA    SSSSSS   CC            TT      RRRRRRRR    LL        OO      OO
BB      BB   AAAAAAAAAA          SS CC            TT      RR   RR     LL        OO      OO
BB      BB   AAAAAAAAAA          SS CC            TT      RR   RR     LL        OO      OO
BB      BB   AA      AA          SS CC            TT      RR    RR    LL        OO      OO
BB      BB   AA      AA          SS CC            TT      RR    RR    LL        OO      OO
BBBBBBBB     AA      AA  SSSSSSSS     CCCCCCCC     TT      RR      RR  LLLLLLLLLL    OOOOOO
BBBBBBBB     AA      AA  SSSSSSSS     CCCCCCCC     TT      RR      RR  LLLLLLLLLL    OOOOOO

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$CTRLO (                          ! Turn control O on or off
;     2          0002  0                 IDENT = '1-004'             ! File: BASCTRLO.B32 Edit: JBS1004
;     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:  VAX-11 BASIC Miscellaneous I/O
;    32          0032  1 !
;    33          0033  1 ! ABSTRACT:
;    34          0034  1 !
;    35          0035  1 !       This module contains the BASIC CTRLO and RCTRLO functions,
;    36          0036  1 !       Which suppress and unsuppress output on a specified channel.
;    37          0037  1 !
;    38          0038  1 ! ENVIRONMENT:  VAX-11 User Mode
;    39          0039  1 !
;    40          0040  1 ! AUTHOR: John Sauter, CREATION DATE: 19-APR-1979
;    41          0041  1 !
;    42          0042  1 ! MODIFIED BY:
;    43          0043  1 !
;    44          0044  1 ! 1-001 - Original.
;    45          0045  1 ! 1-002 - Set up ISB$A_USER_FP.  JBS 25-JUL-1979
;    46          0046  1 ! 1-003 - Correct test of LUB$V_OPENED.  JBS 26-FEB-1980
;    47          0047  1 ! 1-004 - Set CCO bit on the output side of channel 0.  JBS 31-MAR-1980
;    48          0048  1 !--
;    49          0049  1
;    50          0050  1 !<BLF/PAGE>
```

```
;    52          0051  1 !
;    53          0052  1 ! SWITCHES:
;    54          0053  1 !
;    55          0054  1
;    56          0055  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    57          0056  1
;    58          0057  1 !
;    59          0058  1 ! LINKAGES:
;    60          0059  1 !
;    61          0060  1
;    62          0061  1 REQUIRE 'RTLIN:OTSLNK';                         ! Define linkages
;    63          0490  1
;    64          0491  1 !
;    65          0492  1 ! TABLE OF CONTENTS:
;    66          0493  1 !
;    67          0494  1
;    68          0495  1 FORWARD ROUTINE
;    69          0496  1     BAS$CTRLO,                                  ! Suppress output
;    70          0497  1     BAS$RCTRLO;                                 ! Cease suppressing output
;    71          0498  1
;    72          0499  1 !
;    73          0500  1 ! INCLUDE FILES:
;    74          0501  1 !
;    75          0502  1
;    76          0503  1 REQUIRE 'RTLML:OTSLUB';                         ! Get LUB definitions
;    77          0643  1
;    78          0644  1 REQUIRE 'RTLML:OTSISB';                         ! Get ISB definitions
;    79          0812  1
;    80          0813  1 REQUIRE 'RTLIN:RTLPSECT';                       ! Macros for defining psects
;    81          0908  1
;    82          0909  1 LIBRARY 'RTLSTARLE';                            ! System symbols
;    83          0910  1
;    84          0911  1 !
;    85          0912  1 ! MACROS:
;    86          0913  1 !
;    87          0914  1 !       NONE
;    88          0915  1 !
;    89          0916  1 ! EQUATED SYMBOLS:
;    90          0917  1 !
;    91          0918  1 !       NONE
;    92          0919  1 !
;    93          0920  1 ! PSECTS:
;    94          0921  1 !
;    95          0922  1 DECLARE_PSECTS (BAS);                           ! Declare psects for BAS$ facility
;    96          0923  1
;    97          0924  1 ! OWN STORAGE:
;    98          0925  1 !
;    99          0926  1 !       NONE
;   100          0927  1 !
;   101          0928  1 ! EXTERNAL REFERENCES:
;   102          0929  1 !
;   103          0930  1
;   104          0931  1 EXTERNAL ROUTINE
;   105          0932  1     BAS$$OPEN_ZERO : NOVALUE,                   ! Open channel zero
;   106          0933  1     BAS$$CB_PUSH : JSB_CB_PUSH NOVALUE,         ! Load register CCB
;   107          0934  1     BAS$$CB_POP : JSB_CB_POP NOVALUE,           ! Done with register CCB
;   108          0935  1     BAS$$STOP_IO : NOVALUE;                     ! Signal fatal I/O error
```

```
;  109          0936  1
;  110          0937  1 !+
;  111          0938  1 ! The following are the error codes used in this module.
;  112          0939  1 !-
;  113          0940  1
;  114          0941  1 EXTERNAL LITERAL
;  115          0942  1     BAS$K_IO_CHANOT : UNSIGNED (8);             ! Channel not open.
;  116          0943  1
```

```
;  118          0944  1 GLOBAL ROUTINE BAS$CTRLO (                    ! Suppress output
;  119          0945  1         CHAN                                   ! Channel on which to suppress output
;  120          0946  1     ) =
;  121          0947  1
;  122          0948  1 !++
;  123          0949  1 ! FUNCTIONAL DESCRIPTION:
;  124          0950  1 !
;  125          0951  1 !       Simulates typing a control O on the terminal open on the
;  126          0952  1 !       specified channel.
;  127          0953  1 !
;  128          0954  1 ! FORMAL PARAMETERS:
;  129          0955  1 !
;  130          0956  1 !       CHAN.rl.v       The channel whose terminal to simulate a
;  131          0957  1 !                       control O on
;  132          0958  1 !
;  133          0959  1 ! IMPLICIT INPUTS:
;  134          0960  1 !
;  135          0961  1 !       NONE
;  136          0962  1 !
;  137          0963  1 ! IMPLICIT OUTPUTS:
;  138          0964  1 !
;  139          0965  1 !       LUB$V_CCO       Cancel control O.
;  140          0966  1 !
;  141          0967  1 ! ROUTINE VALUE:
;  142          0968  1 ! COMPLETION CODES:
;  143          0969  1 !
;  144          0970  1 !       SS$_NORMAL
;  145          0971  1 !
;  146          0972  1 ! SIDE EFFECTS:
;  147          0973  1 !
;  148          0974  1 !       Signals if an error is encountered.
;  149          0975  1 !       BAS$$CB_PUSH will signal if the channel number is invalid.
;  150          0976  1 !       This function is a no-operation if the channel is not open.
;  151          0977  1 !
;  152          0978  1 !--
;  153          0979  1
;  154          0980  2     BEGIN
;  155          0981  2
;  156          0982  2     BUILTIN
;  157          0983  2         FP;
;  158          0984  2
;  159          0985  2     GLOBAL REGISTER
;  160          0986  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  161          0987  2
;  162          0988  2     LOCAL
;  163          0989  2         FMP : REF BLOCK [, BYTE];
;  164          0990  2
;  165          0991  2     FMP = .FP;
;  166          0992  2 !+
;  167          0993  2 ! Get the CCB for the channel.
;  168          0994  2 !-
;  169          0995  2
;  170          0996  3     IF (.CHAN EQL 0)
;  171          0997  2     THEN
;  172          0998  3         BEGIN
;  173          0999  3 !+
;  174          1000  3 ! The user is referencing his controlling terminal.
```

```
;  175          1001  3 !-
;  176          1002  3       BAS$$CB_PUSH (LUB$K_LUN_BPRI, LUB$K_ILUN_MIN);
;  177          1003  3       CCB [ISB$A_USER_FP] = .FMP [SF$L_SAVE_FP];
;  178          1004  3 !+
;  179          1005  3 ! If the controlling terminal is not yet open, open it.
;  180          1006  3 !-
;  181          1007  3
;  182          1008  3       IF ( NOT .CCB [LUB$V_OPENED]) THEN BAS$$OPEN_ZERO (.FMP [SF$L_SAVE_FP]);
;  183          1009  3
;  184          1010  3       END
;  185          1011  2   ELSE
;  186          1012  2       BEGIN
;  187          1013  3 !+
;  188          1014  3 ! This is an ordinary channel.
;  189          1015  3 !-
;  190          1016  3       BAS$$CB_PUSH (.CHAN, LUB$K_LUN_MIN);
;  191          1017  3       CCB [ISB$A_USER_FP] = .FMP [SF$L_SAVE_FP];
;  192          1018  3       END;
;  193          1019  2
;  194          1020  2 !+
;  195          1021  2 ! If the channel is not now open, this function is a no-operation.
;  196          1022  2 !-
;  197          1023  2
;  198          1024  2   IF (.CCB [LUB$V_OPENED])
;  199          1025  2   THEN
;  200          1026  2       BEGIN
;  201          1027  3 !+
;  202          1028  3 ! Now clear the CCO bit, so control O's will not be canceled.
;  203          1029  3 !-
;  204          1030  3       CCB [LUB$V_CCO] = 0;
;  205          1031  3       END;
;  206          1032  2
;  207          1033  2 !+
;  208          1034  2 ! We are done with register CCB.
;  209          1035  2 !-
;  210          1036  2   BAS$$CB_POP ();
;  211          1037  2   RETURN (SS$_NORMAL);
;  212          1038  1   END;                                   ! end of BAS$CTRLO

                                        .TITLE  BAS$CTRLO
                                        .IDENT  \1-004\

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

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

                                  081C  .ENTRY  BAS$CTRLO, Save R2,R3,R4,R11          ; 0944
54 00000000G   00   9E  00002   MOVAB   BAS$$CB_PUSH, R4                              ;
53             5D   D0  00009   MOVL    FP, FMP                                     ; 0991
         04    AC   D5  0000C   TSTL    CHAN                                        ; 0996
               1E   12  0000F   BNEQ    1$                                          ;
50             08   CE  00011   MNEGL   #8, R0                                      ; 1002
52             08   CE  00014   MNEGL   #8, R2                                      ;
               64   16  00017   JSB     BAS$$CB_PUSH                                ;
```

```
                         FF4C   CB       0C   A3   D0 00019          MOVL    12(FMP), -180(CCB)                      ; 1003
                                1E       FC   AB   E8 0001F          BLBS    -4(CCB), 3$                             ; 1008
                                         0C   A3   DD 00023          PUSHL   12(FMP)                                 ;
                   00000000G     00             01   FB 00026          CALLS   #1, BAS$$OPEN_ZERO                      ;
                                                0E   11   0002D          BRB     2$                                      ; 0996
                                                50   D4   0002F 1$:      CLRL    R0                                      ; 1016
                                52       04   AC   D0 00031          MOVL    CHAN, R2                                ;
                                                64   16   00035          JSB     BAS$$CB_PUSH                            ;
                         FF4C   CB       0C   A3   D0 00037          MOVL    12(FMP), -180(CCB)                      ; 1017
                                04       FC   AB   E9 0003D 2$:      BLBC    -4(CCB), 4$                             ; 1024
                                A0   AB         04   8A   00041 3$:      BICB2   #4, -96(CCB)                            ; 1030
                                   00000000G    00   16   00045 4$:      JSB     BAS$$CB_POP                             ; 1036
                                50              01   D0   0004B          MOVL    #1, R0                                  ; 1037
                                                04   0004E          RET                                             ; 1038

; Routine Size:  79 bytes,    Routine Base:  _BAS$CODE + 0000

;  213          1039  1
```

```
;  215          1040  1 GLOBAL ROUTINE BAS$RCTRLO (                   ! Cancel a typed control O
;  216          1041  1         CHAN                                   ! Channel on which to do this
;  217          1042  1     ) =
;  218          1043  1
;  219          1044  1 !++
;  220          1045  1 ! FUNCTIONAL DESCRIPTION:
;  221          1046  1 !
;  222          1047  1 !       Cancels control O on the terminal open on the specified channel.
;  223          1048  1 !
;  224          1049  1 ! FORMAL PARAMETERS:
;  225          1050  1 !
;  226          1051  1 !       CHAN.rl.v       The channel whose terminal to disable CTRLOing on
;  227          1052  1 !
;  228          1053  1 ! IMPLICIT INPUTS:
;  229          1054  1 !
;  230          1055  1 !       NONE
;  231          1056  1 !
;  232          1057  1 ! IMPLICIT OUTPUTS:
;  233          1058  1 !
;  234          1059  1 !       LUB$V_CCO which, when set, cancels control O.
;  235          1060  1 !
;  236          1061  1 ! ROUTINE VALUE:
;  237          1062  1 ! COMPLETION CODES:
;  238          1063  1 !
;  239          1064  1 !       SS$_NORMAL
;  240          1065  1 !
;  241          1066  1 ! SIDE EFFECTS:
;  242          1067  1 !
;  243          1068  1 !       Signals if an error is encountered.
;  244          1069  1 !       BAS$$CB_PUSH will signal if the channel number is invalid.
;  245          1070  1 !       This routine is a no-operation if the channel is not open.
;  246          1071  1 !
;  247          1072  1 !--
;  248          1073  1
;  249          1074  2     BEGIN
;  250          1075  2
;  251          1076  2     BUILTIN
;  252          1077  2         FP;
;  253          1078  2
;  254          1079  2     GLOBAL REGISTER
;  255          1080  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  256          1081  2
;  257          1082  2     LOCAL
;  258          1083  2         FMP : REF BLOCK [, BYTE];
;  259          1084  2
;  260          1085  2     FMP = .FP;
;  261          1086  2 !+
;  262          1087  2 ! Get the CCB for the channel.
;  263          1088  2 !-
;  264          1089  2
;  265          1090  3     IF (.CHAN EQL 0)
;  266          1091  2     THEN
;  267          1092  3         BEGIN
;  268          1093  3 !+
;  269          1094  3 ! The user is referencing his controlling terminal.
;  270          1095  3 !-
;  271          1096  3         BAS$$CB_PUSH (LUB$K_LUN_BPRI, LUB$K_ILUN_MIN);
```

```
;  272          1097  3          CCB [ISB$A_USER_FP] = .FMP [SF$L_SAVE_FP];
;  273          1098  !+
;  274          1099  ! If the controlling terminal is not yet open, open it.
;  275          1100  !-
;  276          1101
;  277          1102              IF ( NOT .CCB [LUB$V_OPENED]) THEN BAS$$OPEN_ZERO (.FMP [SF$L_SAVE_FP]);
;  278          1103
;  279          1104              END
;  280          1105          ELSE
;  281          1106              BEGIN
;  282          1107  !+
;  283          1108  ! This is an ordinary channel.
;  284          1109  !-
;  285          1110              BAS$$CB_PUSH (.CHAN, LUB$K_LUN_MIN);
;  286          1111              CCB [ISB$A_USER_FP] = .FMP [SF$L_SAVE_FP];
;  287          1112              END;
;  288          1113
;  289          1114  !+
;  290          1115  ! If the channel is not now open, this function is a no-operation.
;  291          1116  !-
;  292          1117
;  293          1118          IF (.CCB [LUB$V_OPENED])
;  294          1119          THEN
;  295          1120              BEGIN
;  296          1121  !+
;  297          1122  ! Now set the CCO bit, which will cause the record level code
;  298          1123  ! to tell RMS to cancel control O.
;  299          1124  !-
;  300          1125              CCB [LUB$V_CCO] = 1;
;  301          1126              END;
;  302          1127
;  303          1128  2 !+
;  304          1129  2 ! We are done with register CCB.
;  305          1130  2 !-
;  306          1131  2     BAS$$CB_POP ();
;  307          1132  2     RETURN (SS$_NORMAL);
;  308          1133  1     END;                                    ! end of BAS$RCTRLO


                                  081C 00000          .ENTRY  BAS$RCTRLO, Save R2,R3,R4,R11                   ; 1040
                    54 00000000G  00   9E 00002        MOVAB   BAS$$CB_PUSH, R4                                ;
                    53            5D   D0 00009        MOVL    FP, FMP                                         ; 1085
                          04      AC   D5 0000C        TSTL    CHAN                                            ; 1090
                                  1E   12 0000F        BNEQ    1$                                              ;
                    50            08   CE 00011        MNEGL   #8, R0                                          ; 1096
                    52            08   CE 00014        MNEGL   #8, R2                                          ;
                                  64   16 00017        JSB     BAS$$CB_PUSH                                    ;
                FF4C  CB    0C    A3   D0 00019        MOVL    12(FMP), -180(CCB)                              ; 1097
                      1E    FC    AB   E8 0001F        BLBS    -4(CCB), 3$                                     ; 1102
                            0C    A3   DD 00023        PUSHL   12(FMP)                                         ;
            00000000G       00         FB 00026        CALLS   #1, BAS$$OPEN_ZERO                              ;
                                  0E   11 0002D        BRB     2$                                              ; 1090
                                  50   D4 0002F  1$:    CLRL    R0                                              ; 1110
                    52            04   AC   D0 00031    MOVL    CHAN, R2                                        ;
```

```
                                 64  16 00035          JSB     BAS$$CB_PUSH                            ;
                      FF4C   CB      0C  A3  D0 00037          MOVL    12(FMP), -180(CCB)                      ; 1111
                             04      FC  AB  E9 0003D 2$:      BLBC    -4(CCB), 4$                             ; 1118
                        A0   AB              04  88 00041 3$:      BISB2   #4, -96(CCB)                    ; 1125
                                 00000000G   00  16 00045 4$:      JSB     BAS$$CB_POP                             ; 1131
                             50              01  D0 0004B          MOVL    #1, R0                          ; 1132
                                             04 0004E            RET                                     ; 1133

; Routine Size:  79 bytes,    Routine Base:  _BAS$CODE + 004F

;  309          1134  1
;  310          1135  1 END                                         ! end of module BAS$CTRLO
;  311          1136  1
;  312          1137  0 ELUDOM



;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  _BAS$CODE                        158  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          2         0       581          00:01.1



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

; Size:         158 code + 0 data bytes
; Run Time:         00:10.0
; Elapsed Time:     00:24.3
; Lines/CPU Min:     6828
; Lexemes/CPU-Min: 40438
; Memory Used:  117 pages
; Compilation Complete
```

BASCLOSE
LIS

BASCONCAT
LIS

BASCTRLO
LIS

BASCHANGE
LIS

BASCTRLC
LIS

BASCHAIN
LIS

BASCOPYFD
LIS

BASCHR
LIS

BASCMPAPP
LIS

BASCUTOUT
LIS

BASCCPOS
LIS