```
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  TTTTTTTTTT  EEEEEEEEEE  RRRRRRRR    MM      MM    IIIIII      000000
BBBBBBBB      AAAAAA      SSSSSSSS  TTTTTTTTTT  EEEEEEEEEE  RRRRRRRR    MM      MM    IIIIII      000000
BB      BB  AA      AA  SS            TT      EE          RR      RR  MMMM  MMMM      II      00      00
BB      BB  AA      AA  SS            TT      EE          RR      RR  MMMM  MMMM      II      00      00
BB      BB  AA      AA  SS            TT      EE          RR      RR  MM  MM  MM      II      00      00
BB      BB  AA      AA  SS            TT      EE          RR      RR  MM  MM  MM      II      00      00
BBBBBBBB    AA      AA    SSSSSS      TT      EEEEEEEE    RRRRRRRR    MM      MM      II      00      00
BBBBBBBB    AA      AA    SSSSSS      TT      EEEEEEEE    RRRRRRRR    MM      MM      II      00      00
BB      BB  AAAAAAAAAA          SS    TT      EE          RR  RR      MM      MM      II      00      00
BB      BB  AAAAAAAAAA          SS    TT      EE          RR  RR      MM      MM      II      00      00
BB      BB  AA      AA          SS    TT      EE          RR    RR    MM      MM      II      00      00
BB      BB  AA      AA          SS    TT      EE          RR    RR    MM      MM      II      00      00
BBBBBBBB    AA      AA  SSSSSSSS      TT      EEEEEEEEEE  RR      RR  MM      MM    IIIIII      000000
BBBBBBBB    AA      AA  SSSSSSSS      TT      EEEEEEEEEE  RR      RR  MM      MM    IIIIII      000000

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$UPI_TERM_IO (                      ! Terminal I/O UPI level
;     2          0002  0                   IDENT = '1-007'             ! File: BASTERMIO.B32 EDIT: PLL1007
;     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 support library - User callable
;    32          0032  1 !
;    33          0033  1 ! ABSTRACT:
;    34          0034  1 !
;    35          0035  1 ! Provides the User Program Interface (UPI) level of support for Basic
;    36          0036  1 ! terminal I/O.
;    37          0037  1 !
;    38          0038  1 ! ENVIRONMENT:  User access mode; AST reentrant
;    39          0039  1 !
;    40          0040  1 ! AUTHOR:       Donald G. Petersen, CREATION DATE: 23-Aug-79
;    41          0041  1 !
;    42          0042  1 ! MODIFIED BY:
;    43          0043  1 !
;    44          0044  1 ! 1-001 - Original done in Macro-32, BASIOELEM.MAR.  DGP 20-Mar-78
;    45          0045  1 ! 1-002 - Rewrite BASIOELEM.MAR in Bliss.  DGP 23-Aug-79
;    46          0046  1 ! 1-003 - Make the BAS$ entry points GLOBAL.  JBS 12-SEP-1979
;    47          0047  1 ! 1-004 - Add BAS$IN_W_R.  JBS 12-SEP-1979
;    48          0048  1 ! 1-004 - Add linkages to CALL of UDF routines.  DGP 12-Sep-79
;    49          0049  1 ! 1-005 - BAS$OUT_D_V_B incorrectly uses a semicolon format character.  DGP
;    50          0050  1 !           17-Sep-79
;    51          0051  1 ! 1-006 - Add support for byte, gfloat, and hfloat.  PLL 24-Aug-81
;    52          0052  1 ! 1-007 - Add support for packed decimal.  PLL 5-Oct-81
;    53          0053  1 !--
;    54          0054  1
;    55          0055  1 !<BLF/PAGE>
```

```
;    57          0056  1 !
;    58          0057  1 ! SWITCHES
;    59          0058  1 !
;    60          0059  1
;    61          0060  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    62          0061  1
;    63          0062  1 !
;    64          0063  1 ! LINKAGES
;    65          0064  1 !
;    66          0065  1
;    67          0066  1 REQUIRE 'RTLIN:BASLNK';                         ! some Basic linkages
;    68          0143  1
;    69          0144  1 REQUIRE 'RTLIN:OTSLNK';                         ! define all other linkages
;    70          0573  1
;    71          0574  1 !
;    72          0575  1 ! TABLE OF CONTENTS:
;    73          0576  1 !
;    74          0577  1
;    75          0578  1 FORWARD ROUTINE
;    76          0579  1     BAS$IN_L_R : NOVALUE,                       ! Input longword by reference
;    77          0580  1     BAS$IN_W_R : NOVALUE,                       ! Input word by reference
;    78          0581  1     BAS$OUT_L_V_S : NOVALUE,                    ! output long by value, semi format char
;    79          0582  1     BAS$OUT_L_V_C : NOVALUE,                    ! output long by val, comma format char
;    80          0583  1     BAS$OUT_L_V_B : NOVALUE,                    ! output long by val, no format char
;    81          0584  1     BAS$IN_F_R : NOVALUE,                       ! input single float by ref
;    82          0585  1     BAS$OUT_F_V_S : NOVALUE,                    ! output float by val, semi format char
;    83          0586  1     BAS$OUT_F_V_C : NOVALUE,                    ! output float by val, comma format char
;    84          0587  1     BAS$OUT_F_V_B : NOVALUE,                    ! output float by val, no format char
;    85          0588  1     BAS$IN_D_R : NOVALUE,                       ! input double float by ref
;    86          0589  1     BAS$OUT_D_V_S : NOVALUE,                    ! output double by val, semi format char
;    87          0590  1     BAS$OUT_D_V_C : NOVALUE,                    ! output double by val, comma format char
;    88          0591  1     BAS$OUT_D_V_B : NOVALUE,                    ! output double by val, no format char
;    89          0592  1     BAS$IN_T_DX : NOVALUE,                      ! input text by descriptor
;    90          0593  1     BAS$OUT_T_DX_S : NOVALUE,                   ! output text by desc, semi format char
;    91          0594  1     BAS$OUT_T_DX_C : NOVALUE,                   ! output text by desc, comma format char
;    92          0595  1     BAS$OUT_T_DX_B : NOVALUE,                   ! output text by desc, no format char
;    93          0596  1     BAS$OUT_G_V_S : NOVALUE,                    ! output G by value, semi format
;    94          0597  1     BAS$OUT_G_V_C : NOVALUE,                    ! output G by value, comma format
;    95          0598  1     BAS$OUT_G_V_B : NOVALUE,                    ! output G by value, no format
;    96          0599  1     BAS$OUT_H_V_S : NOVALUE,                    ! output H by value, semi format
;    97          0600  1     BAS$OUT_H_V_C : NOVALUE,                    ! output H by value, comma format
;    98          0601  1     BAS$OUT_H_V_B : NOVALUE,                    ! output H by value, no format
;    99          0602  1     BAS$OUT_P_DX_S : NOVALUE,                   ! output packed by desc, semi format
;   100          0603  1     BAS$OUT_P_DX_C : NOVALUE,                   ! output packed by desc, comma format
;   101          0604  1     BAS$OUT_P_DX_B : NOVALUE,                   ! output packed by desc, no format
;   102          0605  1     BAS$IN_G_R : NOVALUE,                       ! input G by ref
;   103          0606  1     BAS$IN_H_R : NOVALUE,                       ! input H by ref
;   104          0607  1     BAS$IN_B_R : NOVALUE,                       ! input byte by ref
;   105          0608  1     BAS$IN_P_DX : NOVALUE;                      ! input packed by desc
;   106          0609  1
;   107          0610  1 !
;   108          0611  1 ! INCLUDE FILES:
;   109          0612  1 !
;   110          0613  1
;   111          0614  1 REQUIRE 'RTLIN:RTLPSECT';                       ! PSECT macros
;   112          0709  1
;   113          0710  1 REQUIRE 'RTLML:BASPAR';                         ! Some Basic symbols
```

```
;  114          0732  1
;  11           0733  1 REQUIRE 'RTLML:OTSISB';                    ! I/O statement block (ISB)
;    6          0901  1
;  117          0902  1 REQUIRE 'RTLML:OTSLUB';                   ! logical unit block (LUB)
   118          1042  1
   119          1043  1 LIBRARY 'RTLSTARLE';                       ! STARTLET library for macros
   120          1044  1
;  121          1045  1 !
;  122          1046  1 ! MACROS:
;  123          1047  1 !
;  124          1048  1 !       NONE
;  125          1049  1 !
;  126          1050  1 ! PSECT DECLARATIONS:
;  127          1051  1 !
;  128          1052  1 DECLARE_PSECTS (BAS);                     ! PSECTs for BAS$ facility
;  129          1053  1 !
;  130          1054  1 ! EQUATED SYMBOLS:
;  131          1055  1 !
;  132          1056  1
;  133          1057  1 LITERAL
;  134          1058  1     K_BYTE_LEN = 1,                      ! storage size in bytes of byte
;  135          1059  1     K_WORD_LEN = 2,                      ! storage size in bytes of word
;  136          1060  1     K_LONG_LEN = 4,                      ! storage size in bytes of longword
;  137          1061  1     K_FLOAT_LEN = 4,                     ! storage size in bytes of float
;  138          1062  1     K_DOUBLE_LEN = 8,                    ! storage size in bytes of double
;  139          1063  1     K_GFLOAT_LEN = 8,                    ! storage size in bytes of gfloat
;  140          1064  1     K_HFLOAT_LEN = 16;                   ! storage size in bytes of hfloat
;  141          1065  1
;  142          1066  1 !
;  143          1067  1 ! OWN STORAGE:
;  144          1068  1 !
;  145          1069  1 !       NONE
;  146          1070  1 !
;  147          1071  1 ! EXTERNAL REFERENCES:
;  148          1072  1 !
;  149          1073  1
;  150          1074  1 EXTERNAL ROUTINE
;  151          1075  1     BAS$$UDF_RL1 : CALL_CCB,             ! UDF level element transmitter - Read
;  152          1076  1                                          ! list directed
;  153          1077  1     BAS$$STOP : NOVALUE;                 ! signal an error and stop
;  154          1078  1
;  155          1079  1 EXTERNAL LITERAL
;  156          1080  1     BAS$K_DATTYPERR;                     ! data type error
;  157          1081  1
;  158          1082  1 EXTERNAL
;  159          1083  1     OTS$$A_CUR_LUB,                      ! Contains addr. of current LUB/ISB/RAB
;  160          1084  1     BAS$$AA_UDF_PR1 : VECTOR;            ! dispatch vector to UDF element transmitters
;  161          1085  1
```

```
;  163          1086  1 GLOBAL ROUTINE BAS$IN_L_R (                    ! input longword by ref
;  164          1087  1         ELEM                                    ! element to input by reference
;  165          1088  1     ) : NOVALUE =                               !
;  166          1089  1
;  167          1090  1 !++
;  168          1091  1 ! FUNCTIONAL DESCRIPTION:
;  169          1092  1 !
;  170          1093  1 ! Input a longword, the destination is passed by reference
;  171          1094  1 !
;  172          1095  1 ! FORMAL PARAMETERS:
;  173          1096  1 !
;  174          1097  1 !       ELEM.rl.r               where to store the longword input
;  175          1098  1 !
;  176          1099  1 ! IMPLICIT INPUTS:
;  177          1100  1 !
;  178          1101  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  179          1102  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  180          1103  1 !
;  181          1104  1 ! IMPLICIT OUTPUTS:
;  182          1105  1 !
;  183          1106  1 !       NONE
;  184          1107  1 !
;  185          1108  1 ! COMPLETION CODES:
;  186          1109  1 !
;  187          1110  1 !       NONE
;  188          1111  1 !
;  189          1112  1 ! SIDE EFFECTS:
;  190          1113  1 !
;  191          1114  1 !       NONE
;  192          1115  1 !
;  193          1116  1 !--
;  194          1117  1
;  195          1118  2     BEGIN
;  196          1119  2
;  197          1120  2     GLOBAL REGISTER
;  198          1121  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  199          1122  2
;  200          1123  2     CCB = .OTS$$A_CUR_LUB;
;  201          1124  2     BAS$$UDF_RL1 (DSC$K_DTYPE_L,                ! data type longword
;  202          1125  2         K_LONG_LEN,                             ! length of data type
;  203          1126  2         .ELEM,                                  ! address of destination
;  204          1127  2         BAS$K_NULL);                            ! null format character
;  205          1128  2     RETURN;
;  206          1129  1     END;                                        !End of BAS$IN_L_R


                                        .TITLE  BAS$UPI_TERM_IO
                                        .IDENT  \1-007\

                                        .EXTRN  BAS$$UDF_RL1, BAS$$STOP
                                        .EXTRN  BAS$K_DATTYPERR
                                        .EXTRN  OTS$$A_CUR_LUB, BAS$$AA_UDF_PR1

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

                            0800 00000  .ENTRY  BAS$IN_L_R, Save R11                    ; 1086
5B 00000000G   00   D0 00002            MOVL    OTS$$A_COR_LUB, CCB                 ; 1123
```

```
                                    7E  D4 00009          CLRL    -(SP)                                  ; 1124
                                04  AC  DD 0000B          PUSHL   ELEM                                   ; 1126
                                    04  DD 0000E          PUSHL   #4                                     ; 1124
                                    08  DD 00010          PUSHL   #8                                     ;
                    00000000G   00      04  FB 00012      CALLS   #4, BAS$$UDF_RL1                       ;
                                        04 00019          RET                                            ; 1129

; Routine Size:  26 bytes,    Routine Base:  _BAS$CODE + 0000

;  207          1130  1
```

```
;  209          1131  1 GLOBAL ROUTINE BAS$IN_W_R (                    ! input word by ref
;  210          1132  1         ELEM                                    ! element to input by reference
;  211          1133  1     ) : NOVALUE =                               !
;  212          1134  1
;  213          1135  1 !++
;  214          1136  1 ! FUNCTIONAL DESCRIPTION:
;  215          1137  1 !
;  216          1138  1 ! Input a word, the destination is passed by reference
;  217          1139  1 !
;  218          1140  1 ! FORMAL PARAMETERS:
;  219          1141  1 !
;  220          1142  1 !       ELEM.ww.r               where to store the longword input
;  221          1143  1 !
;  222          1144  1 ! IMPLICIT INPUTS:
;  223          1145  1 !
;  224          1146  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  225          1147  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  226          1148  1 !
;  227          1149  1 ! IMPLICIT OUTPUTS:
;  228          1150  1 !
;  229          1151  1 !       NONE
;  230          1152  1 !
;  231          1153  1 ! COMPLETION CODES:
;  232          1154  1 !
;  233          1155  1 !       NONE
;  234          1156  1 !
;  235          1157  1 ! SIDE EFFECTS:
;  236          1158  1 !
;  237          1159  1 !       NONE
;  238          1160  1 !
;  239          1161  1 !--
;  240          1162  1
;  241          1163  2     BEGIN
;  242          1164  2
;  243          1165  2     GLOBAL REGISTER
;  244          1166  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  245          1167  2
;  246          1168  2     CCB = .OTS$$A_CUR_LUB;
;  247          1169  2     BAS$$UDF_RL1 (DSC$K_DTYPE_W,                ! data type word
;  248          1170  2         K_WORD_LEN,                             ! length of data type
;  249          1171  2         .ELEM,                                  ! address of destination
;  250          1172  2         BAS$K_NULL);                            ! null format character
;  251          1173  2     RETURN;
;  252          1174  1     END;                                        !End of BAS$IN_W_R


                                  0800 00000         .ENTRY  BAS$IN_W_R, Save R11                    ; 1131
                5B 00000000G   00   D0 00002         MOVL    OTS$$A_CUR_LUB, CCB                     ; 1168
                               7E   D4 00009         CLRL    -(SP)                                   ; 1169
                         04    AC   DD 0000B         PUSHL   ELEM                                    ; 1171
                               02   DD 0000E         PUSHL   #2                                      ; 1169
                               07   DD 00010         PUSHL   #7                                      ;
                00000000G      00   FB 00012         CALLS   #4, BAS$$UDF_RL1                        ;
                                    04 00019         RET                                             ; 1174
```

```
; Routine Size:  26 bytes,    Routine Base:  _BAS$CODE + 001A

;  253          1175  1
```

```
;  255          1176  1 GLOBAL ROUTINE BAS$IN_F_R (                          ! input float by ref
;  256          1177  1         ELEM                                         ! element to input by reference
;  257          1178  1     ) : NOVALUE =                                    !
;  258          1179  1
;  259          1180  1 !++
;  260          1181  1 ! FUNCTIONAL DESCRIPTION:
;  261          1182  1 !
;  262          1183  1 ! Input a single precision floating, the destination is passed by reference
;  263          1184  1 !
;  264          1185  1 ! FORMAL PARAMETERS:
;  265          1186  1 !
;  266          1187  1 !       ELEM.rf.r               where to store the float input
;  267          1188  1 !
;  268          1189  1 ! IMPLICIT INPUTS:
;  269          1190  1 !
;  270          1191  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  271          1192  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  272          1193  1 !
;  273          1194  1 ! IMPLICIT OUTPUTS:
;  274          1195  1 !
;  275          1196  1 !       NONE
;  276          1197  1 !
;  277          1198  1 ! COMPLETION CODES:
;  278          1199  1 !
;  279          1200  1 !       NONE
;  280          1201  1 !
;  281          1202  1 ! SIDE EFFECTS:
;  282          1203  1 !
;  283          1204  1 !       NONE
;  284          1205  1 !
;  285          1206  1 !--
;  286          1207  1
;  287          1208  2     BEGIN
;  288          1209  2
;  289          1210  2     GLOBAL REGISTER
;  290          1211  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  291          1212  2
;  292          1213  2     CCB = .OTS$$A_CUR_LUB;
;  293          1214  2     BAS$$UDF_RL1 (DSC$K_DTYPE_F,                       ! data type floating
;  294          1215  2         K_FLOAT_LEN,                                   ! length of data type
;  295          1216  2         .ELEM,                                         ! address of destination
;  296          1217  2         BAS$K_NULL);                                   ! null format character
;  297          1218  2     RETURN;
;  298          1219  1     END;                                               !End of BAS$IN_F_R


                                  0800 00000         .ENTRY  BAS$IN_F_R, Save R11                    ; 1176
                5B 00000000G   00   D0 00002         MOVL    OTS$$A_CUR_LUB, CCB                     ; 1213
                               7E   D4 00009         CLRL    -(SP)                                   ; 1214
                         04    AC   DD 0000B         PUSHL   ELEM                                    ; 1216
                               04   DD 0000E         PUSHL   #4                                      ; 1214
                               0A   DD 00010         PUSHL   #10                                     ;
                00000000G      00   FB 00012         CALLS   #4, BAS$$UDF_RL1                        ;
                                    04 00019         RET                                             ; 1219
```

```
; Routine Size:  26 bytes,    Routine Base:  _BAS$CODE + 0034

;  299          1220  1
```

```
;  301          1221  1 GLOBAL ROUTINE BAS$IN_D_R (                          ! input double by ref
;  302          1222  1         ELEM                                         ! element to input by reference
;  303          1223  1     ) : NOVALUE =                                    !
;  304          1224  1
;  305          1225  1 !++
;  306          1226  1 ! FUNCTIONAL DESCRIPTION:
;  307          1227  1 !
;  308          1228  1 ! Input a double, the destination is passed by reference
;  309          1229  1 !
;  310          1230  1 ! FORMAL PARAMETERS:
;  311          1231  1 !
;  312          1232  1 !       ELEM.rd.r               where to store the double input
;  313          1233  1 !
;  314          1234  1 ! IMPLICIT INPUTS:
;  315          1235  1 !
;  316          1236  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  317          1237  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  318          1238  1 !
;  319          1239  1 ! IMPLICIT OUTPUTS:
;  320          1240  1 !
;  321          1241  1 !       NONE
;  322          1242  1 !
;  323          1243  1 ! COMPLETION CODES:
;  324          1244  1 !
;  325          1245  1 !       NONE
;  326          1246  1 !
;  327          1247  1 ! SIDE EFFECTS:
;  328          1248  1 !
;  329          1249  1 !       NONE
;  330          1250  1 !
;  331          1251  1 !--
;  332          1252  1
;  333          1253  2     BEGIN
;  334          1254  2
;  335          1255  2     GLOBAL REGISTER
;  336          1256  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  337          1257  2
;  338          1258  2     CCB = .OTS$$A_CUR_LUB;
;  339          1259  2     BAS$$UDF_RL1 (DSC$K_DTYPE_D,                       ! data type
;  340          1260  2         K_DOUBLE_LEN,                                  ! length of data type
;  341          1261  2         .ELEM,                                         ! address of destination
;  342          1262  2         BAS$K_NULL);                                   ! null format character
;  343          1263  2     RETURN;
;  344          1264  1     END;                                               !End of BAS$IN_D_R


                                    0800 00000          .ENTRY  BAS$IN_D_R, Save R11                    ; 1221
                         5B 00000000G   00   D0 00002   MOVL    OTS$$A_CUR_LUB, CCB                     ; 1258
                                        7E   D4 00009   CLRL    -(SP)                                   ; 1259
                                  04    AC   DD 0000B   PUSHL   ELEM                                    ; 1261
                                        08   DD 0000E   PUSHL   #8                                      ; 1259
                                        0B   DD 00010   PUSHL   #11                                     ;
                00000000G   00          04   FB 00012   CALLS   #4, BAS$$UDF_RL1                        ;
                                             04 00019   RET                                             ; 1264
```

```
; Routine Size:  26 bytes,    Routine Base:  _BAS$CODE + 004E

;  345          1265  1
```

```
1 GLOBAL ROUTINE BAS$IN_T_DX (                        ! input text by desc.
1           ELEM                                      ! element to input by reference
1     ) : NOVALUE =                                   !
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 ! Input text, the destination is passed by descriptor
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !         ELEM.rt.dx              where to store the text input
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !         OTS$$A_CUR_LUB          current Logical Unit Block
1 !         ISB$B_STTM_TYPE         statement type of this I/O statement
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !         NONE
1 !
1 ! COMPLETION CODES:
1 !
1 !         NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !         NONE
1 !
1 !--
1
2     BEGIN
2
2     GLOBAL REGISTER
2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
2
2     MAP
2         ELEM : REF BLOCK [8, BYTE];
2
2     CCB = .OTS$$A_CUR_LUB;
2     BAS$$UDF_RL1 (DSC$K_DTYPE_T,                    ! data type
2         ELEM [DSC$W_LENGTH],                        ! string length
2         .ELEM,                                      ! address of descriptor
2         BAS$K_NULL);                                ! null format character
2     RETURN;
1     END;                                            !End of BAS$IN_T_DX
```

```
                       0800 00000     .ENTRY  BAS$IN_T_DX, Save R11          ; 1266
  5B 00000000G  00  D0 00002     MOVL    OTS$$A_CUR_LUB, CCB            ; 1306
                7E  D4 00009     CLRL    -(SP)                          ; 1308
       04       AC  DD 0000B     PUSHL   ELEM                           ; 1309
       04       AC  DD 0000E     PUSHL   ELEM                           ; 1308
```

```
                                      0E  DD 00011          PUSHL   #14                              ;
                00000000G  00         04  FB 00013          CALLS   #4, BAS$$UDF_RL1                 ;
                                          04 0001A          RET                                      ; 1312

; Routine Size:  27 bytes,    Routine Base:  _BAS$CODE + 0068


;  394          1313  1
```

```
;  396          1314  1 GLOBAL ROUTINE BAS$OUT_L_V_S (                    ! print longword by value, semi format character
;  397          1315  1         ELEM                                      ! element to print by value
;  398          1316  1     ) : NOVALUE =                                 !
;  399          1317  1
;  400          1318  1 !++
;  401          1319  1 ! FUNCTIONAL DESCRIPTION:
;  402          1320  1 !
;  403          1321  1 ! Output a longword, the source is passed by value
;  404          1322  1 !
;  405          1323  1 ! FORMAL PARAMETERS:
;  406          1324  1 !
;  407          1325  1 !       ELEM.rl.v               longword to print
;  408          1326  1 !
;  409          1327  1 ! IMPLICIT INPUTS:
;  410          1328  1 !
;  411          1329  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  412          1330  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  413          1331  1 !
;  414          1332  1 ! IMPLICIT OUTPUTS:
;  415          1333  1 !
;  416          1334  1 !       NONE
;  417          1335  1 !
;  418          1336  1 ! COMPLETION CODES:
;  419          1337  1 !
;  420          1338  1 !       NONE
;  421          1339  1 !
;  422          1340  1 ! SIDE EFFECTS:
;  423          1341  1 !
;  424          1342  1 !       NONE
;  425          1343  1 !
;  426          1344  1 !--
;  427          1345  1
;  428          1346  2     BEGIN
;  429          1347  2
;  430          1348  2     GLOBAL REGISTER
;  431          1349  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  432          1350  2
;  433          1351  2     CCB = .OTS$$A_CUR_LUB;
;  434          1352  2 !+
;  435          1353  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  436          1354  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  437          1355  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  438          1356  2 !-
;  439          1357  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_L,
;  440          1358  2                                                     ! data type
;  441          1359  2         K_LONG_LEN,                                 ! length of data type
;  442          1360  2         ELEM,                                       ! address of value
;  443          1361  2         BAS$K_SEMI_FORM                             ! semicolon format character
;  444          1362  2     );
;  445          1363  2     RETURN;
;  446          1364  1     END;                                            !End of BAS$OUT_L_V_S


                                0800 00000          .ENTRY  BAS$OUT_L_V_S, Save R11                         ; 1314
```

```
                    5B 00000000G   00   DO 00002          MOVL    OTS$$A_CUR_LUB, CCB                          ; 1351
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                                ; 1357
                    50 00000000G0040   DO 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0                  ;
                                   01   DD 00016          PUSHL   #1                                           ;
                                04    AC   9F 00018       PUSHAB  ELEM                                         ;
                                   04   DD 0001B          PUSHL   #4                                           ;
                                   08   DD 0001D          PUSHL   #8                                           ;
                000000000G0040     04   FB 0001F          CALLS   #4, BAS$$AA_UDF_PR1[R0]                      ;
                                   04 00027              RET                                                  ; 1364

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 0083

;  447          1365  1
```

```
;  449          1366  1 GLOBAL ROUTINE BAS$OUT_L_V_C (                    ! print a longword by value, comma format character
;  450          1367  1         ELEM                                      ! element to output by value
;  451          1368  1     ) : NOVALUE =                                 !
;  452          1369  1
;  453          1370  1 !++
;  454          1371  1 ! FUNCTIONAL DESCRIPTION:
;  455          1372  1 !
;  456          1373  1 ! Output a longword, the source is passed by value
;  457          1374  1 !
;  458          1375  1 ! FORMAL PARAMETERS:
;  459          1376  1 !
;  460          1377  1 !       ELEM.rl.v               longword to print
;  461          1378  1 !
;  462          1379  1 ! IMPLICIT INPUTS:
;  463          1380  1 !
;  464          1381  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  465          1382  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  466          1383  1 !
;  467          1384  1 ! IMPLICIT OUTPUTS:
;  468          1385  1 !
;  469          1386  1 !       NONE
;  470          1387  1 !
;  471          1388  1 ! COMPLETION CODES:
;  472          1389  1 !
;  473          1390  1 !       NONE
;  474          1391  1 !
;  475          1392  1 ! SIDE EFFECTS:
;  476          1393  1 !
;  477          1394  1 !       NONE
;  478          1395  1 !
;  479          1396  1 !--
;  480          1397  1
;  481          1398  2     BEGIN
;  482          1399  2
;  483          1400  2     GLOBAL REGISTER
;  484          1401  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  485          1402  2
;  486          1403  2     CCB = .OTS$$A_CUR_LUB;
;  487          1404  2 !+
;  488          1405  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  489          1406  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  490          1407  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  491          1408  2 !-
;  492          1409  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_L,
;  493          1410  2                                                  ! data type
;  494          1411  2         K_LONG_LEN,                              ! length of data type
;  495          1412  2         ELEM,                                    ! address of value
;  496          1413  2         BAS$K_COMMA_FOR                          ! comma format character
;  497          1414  2     );
;  498          1415  2     RETURN;
;  499          1416  1     END;                                         !End of BAS$OUT_L_V_C


                                0800 00000          .ENTRY  BAS$OUT_L_V_C, Save R11                         ; 1366
```

```
                    5B 00000000G   00   DO 00002        MOVL    OTS$$A_CUR_LUB, CCB                     ; 1403
                    50      FF71   CB   9A 00009        MOVZBL  -143(CCB), R0                           ; 1409
                    50 00000000G0040   DO 0000E        MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                                   02   DD 00016        PUSHL   #2                                      ;
                                04 AC   9F 00018        PUSHAB  ELEM                                    ;
                                   04   DD 0001B        PUSHL   #4                                      ;
                                   08   DD 0001D        PUSHL   #8                                      ;
                000000000G0040         04   FB 0001F    CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                           04 00027     RET                                             ; 1416

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 00AB

;  500          1417  1
```

```
;  502          1418  1 GLOBAL ROUTINE BAS$OUT_L_V_B (                    ! print longword by value, no format character
;  503          1419  1         ELEM                                      ! element to output by value
;  504          1420  1     ) : NOVALUE =                                 !
;  505          1421  1
;  506          1422  1 !++
;  507          1423  1 ! FUNCTIONAL DESCRIPTION:
;  508          1424  1 !
;  509          1425  1 ! Output a longword, the source is passed by value
;  510          1426  1 !
;  511          1427  1 ! FORMAL PARAMETERS:
;  512          1428  1 !
;  513          1429  1 !       ELEM.rl.v               longword to print
;  514          1430  1 !
;  515          1431  1 ! IMPLICIT INPUTS:
;  516          1432  1 !
;  517          1433  1 !       .OTS$$A_CUR_LUB         current Logical Unit Block
;  518          1434  1 !        ISB$B_STTM_TYPE        statement type of this I/O statement
;  519          1435  1 !
;  520          1436  1 ! IMPLICIT OUTPUTS:
;  521          1437  1 !
;  522          1438  1 !       NONE
;  523          1439  1 !
;  524          1440  1 ! COMPLETION CODES:
;  525          1441  1 !
;  526          1442  1 !       NONE
;  527          1443  1 !
;  528          1444  1 ! SIDE EFFECTS:
;  529          1445  1 !
;  530          1446  1 !       NONE
;  531          1447  1 !
;  532          1448  1 !--
;  533          1449  1
;  534          1450  2     BEGIN
;  535          1451  2
;  536          1452  2     GLOBAL REGISTER
;  537          1453  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  538          1454  2
;  539          1455  2     CCB = .OTS$$A_CUR_LUB;
;  540          1456  2 !+
;  541          1457  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  542          1458  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  543          1459  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  544          1460  2 !-
;  545          1461  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_L,
;  546          1462  2                                                  ! data type
;  547          1463  2         K_LONG_LEN,                              ! length of data type
;  548          1464  2         ELEM,                                    ! address of value
;  549          1465  2         BAS$K_NO_FORM                            ! no format character
;  550          1466  2     );
;  551          1467  2     RETURN;
;  552          1468  1     END;                                         !End of BAS$OUT_L_V_B


                                0800 00000          .ENTRY  BAS$OUT_L_V_B, Save R11                 ; 1418
```

```
                          5B 00000000G   00   DO 00002          MOVL    OTS$$A_CUR_LUB, CCB                          ; 1455
                          50     FF71    CB   9A 00009          MOVZBL  -143(CCB), R0                                ; 1461
                          50 00000000G0040   DO 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0                  ;
                                         03   DD 00016          PUSHL   #3                                           ;
                                     04  AC   9F 00018          PUSHAB  ELEM                                         ;
                                         04   DD 0001B          PUSHL   #4                                           ;
                                         08   DD 0001D          PUSHL   #8                                           ;
                    000000000G0040       04   FB 0001F          CALLS   #4, BAS$$AA_UDF_PR1[R0]                      ;
                                              04 00027          RET                                                  ; 1468

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 00D3

;  553          1469  1
```

```
;  555          1470  1 GLOBAL ROUTINE BAS$OUT_F_V_S (                    ! print floating by value, semicolon format
;  556          1471  1         ELEM                                      ! element to output by value
;  557          1472  1     ) : NOVALUE =                                 !
;  558          1473  1
;  559          1474  1 !++
;  560          1475  1 ! FUNCTIONAL DESCRIPTION:
;  561          1476  1 !
;  562          1477  1 ! Output a single precision floating, the source is passed by value
;  563          1478  1 !
;  564          1479  1 ! FORMAL PARAMETERS:
;  565          1480  1 !
;  566          1481  1 !       ELEM.rf.v               floating to print
;  567          1482  1 !
;  568          1483  1 ! IMPLICIT INPUTS:
;  569          1484  1 !
;  570          1485  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  571          1486  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  572          1487  1 !
;  573          1488  1 ! IMPLICIT OUTPUTS:
;  574          1489  1 !
;  575          1490  1 !       NONE
;  576          1491  1 !
;  577          1492  1 ! COMPLETION CODES:
;  578          1493  1 !
;  579          1494  1 !       NONE
;  580          1495  1 !
;  581          1496  1 ! SIDE EFFECTS:
;  582          1497  1 !
;  583          1498  1 !       NONE
;  584          1499  1 !
;  585          1500  1 !--
;  586          1501  1
;  587          1502  2     BEGIN
;  588          1503  2
;  589          1504  2     GLOBAL REGISTER
;  590          1505  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  591          1506  2
;  592          1507  2     CCB = .OTS$$A_CUR_LUB;
;  593          1508  2 !+
;  594          1509  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  595          1510  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  596          1511  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  597          1512  2 !-
;  598          1513  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_F,
;  599          1514  2                                                  ! data type
;  600          1515  2         K_FLOAT_LEN,                             ! length of data type
;  601          1516  2         ELEM,                                    ! address of value
;  602          1517  2         BAS$K_SEMI_FORM                          ! semicolon format character
;  603          1518  2     );
;  604          1519  2     RETURN;
;  605          1520  1     END;                                         !End of BAS$OUT_F_V_S


                             0800 00000         .ENTRY  BAS$OUT_F_V_S, Save R11                         ; 1470
```

```
                    5B 00000000G   00   D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ; 1507
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                           ; 1513
                    50 00000000G0040   D0 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                                   01   DD 00016          PUSHL   #1                                      ;
                                   04   AC 9F 00018       PUSHAB  ELEM                                    ;
                                   04   DD 0001B          PUSHL   #4                                      ;
                                   0A   DD 0001D          PUSHL   #10                                     ;
                    000000000G0040     04   FB 0001F     CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                       04 00027          RET                                             ; 1520

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 00FB

;  606          1521  1
```

```
;  608          1522  1 GLOBAL ROUTINE BAS$OUT_F_V_C (                        ! print floating by value, comma format
;  609          1523  1         ELEM                                          ! element to output by value
;  610          1524  1     ) : NOVALUE =                                     !
;  611          1525  1
;  612          1526  1 !++
;  613          1527  1 ! FUNCTIONAL DESCRIPTION:
;  614          1528  1 !
;  615          1529  1 ! Output a single precision floating, the source is passed by value
;  616          1530  1 !
;  617          1531  1 ! FORMAL PARAMETERS:
;  618          1532  1 !
;  619          1533  1 !       ELEM.rf.v               floating to print
;  620          1534  1 !
;  621          1535  1 ! IMPLICIT INPUTS:
;  622          1536  1 !
;  623          1537  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  624          1538  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  625          1539  1 !
;  626          1540  1 ! IMPLICIT OUTPUTS:
;  627          1541  1 !
;  628          1542  1 !       NONE
;  629          1543  1 !
;  630          1544  1 ! COMPLETION CODES:
;  631          1545  1 !
;  632          1546  1 !       NONE
;  633          1547  1 !
;  634          1548  1 ! SIDE EFFECTS:
;  635          1549  1 !
;  636          1550  1 !       NONE
;  637          1551  1 !
;  638          1552  1 !--
;  639          1553  1
;  640          1554  2     BEGIN
;  641          1555  2
;  642          1556  2     GLOBAL REGISTER
;  643          1557  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  644          1558  2
;  645          1559  2     CCB = .OTS$$A_CUR_LUB;
;  646          1560  2 !+
;  647          1561  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  648          1562  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  649          1563  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  650          1564  2 !-
;  651          1565  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_F,
;  652          1566  2                                                         ! data type
;  653          1567  2         K_FLOAT_LEN,                                    ! length of data type
;  654          1568  2         ELEM,                                           ! address of value
;  655          1569  2         BAS$K_COMMA_FOR                                 ! comma format character
;  656          1570  2     );
;  657          1571  2     RETURN;
;  658          1572  1     END;                                               !End of BAS$OUT_F_V_C


                                0800 00000         .ENTRY  BAS$OUT_F_V_C, Save R11                         ; 1522
```

```
                    5B 00000000G   00   D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ; 1559
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                           ; 1565
                    50 00000000G0040   D0 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                                   02   DD 00016          PUSHL   #2                                      ;
                                   04   AC 9F 00018       PUSHAB  ELEM                                    ;
                                   04   DD 0001B          PUSHL   #4                                      ;
                                   0A   DD 0001D          PUSHL   #10                                     ;
                00000000G0040      04   FB 0001F          CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                        04 00027          RET                                             ; 1572

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 0123

;  659          1573  1
```

```
;  661          1574  1 GLOBAL ROUTINE BAS$OUT_F_V_B (                        ! print floating by value, no format
;  662          1575  1         ELEM                                          ! element to output by value
;  663          1576  1     ) : NOVALUE =                                     !
;  664          1577  1
;  665          1578  1 !++
;  666          1579  1 ! FUNCTIONAL DESCRIPTION:
;  667          1580  1 !
;  668          1581  1 ! Output a single precision floating, the source is passed by value
;  669          1582  1 !
;  670          1583  1 ! FORMAL PARAMETERS:
;  671          1584  1 !
;  672          1585  1 !       ELEM.rf.v               floating to print
;  673          1586  1 !
;  674          1587  1 ! IMPLICIT INPUTS:
;  675          1588  1 !
;  676          1589  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  677          1590  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  678          1591  1 !
;  679          1592  1 ! IMPLICIT OUTPUTS:
;  680          1593  1 !
;  681          1594  1 !       NONE
;  682          1595  1 !
;  683          1596  1 ! COMPLETION CODES:
;  684          1597  1 !
;  685          1598  1 !       NONE
;  686          1599  1 !
;  687          1600  1 ! SIDE EFFECTS:
;  688          1601  1 !
;  689          1602  1 !       NONE
;  690          1603  1 !
;  691          1604  1 !--
;  692          1605  1
;  693          1606  2     BEGIN
;  694          1607  2
;  695          1608  2     GLOBAL REGISTER
;  696          1609  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  697          1610  2
;  698          1611  2     CCB = .OTS$$A_CUR_LUB;
;  699          1612  2 !+
;  700          1613  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  701          1614  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  702          1615  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  703          1616  2 !-
;  704          1617  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_F,
;  705          1618  2                                                         ! data type
;  706          1619  2         K_FLOAT_LEN,                                    ! length of data type
;  707          1620  2         ELEM,                                           ! address of value
;  708          1621  2         BAS$K_NO_FORM                                   ! no format character
;  709          1622  2     );
;  710          1623  2     RETURN;
;  711          1624  1     END;                                                !End of BAS$OUT_F_V_B


                                0800 00000          .ENTRY  BAS$OUT_F_V_B, Save R11                         ; 1574
```

```
                    5B 00000000G   00   DO 00002        MOVL    OTS$$A_CUR_LUB, CCB                     ; 1611
                    50      FF71   CB   9A 00009        MOVZBL  -143(CCB), R0                           ; 1617
                    50 00000000G0040   DO 0000E        MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                                   03   DD 00016        PUSHL   #3                                      ;
                                04    AC   9F 00018        PUSHAB  ELEM                                    ;
                                   04   DD 0001B        PUSHL   #4                                      ;
                                   0A   DD 0001D        PUSHL   #10                                     ;
                00000000G0040          04   FB 0001F        CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                          04 00027        RET                                             ; 1624

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 014B

;  712          1625  1
```

```
;  714          1626  1 GLOBAL ROUTINE BAS$OUT_D_V_S (                        ! print double by value, semicolon format
;  715          1627  1         ELEM                                          ! element to output by value
;  716          1628  1     ) : NOVALUE =                                     .
;  717          1629  1
;  718          1630  1 !++
;  719          1631  1 ! FUNCTIONAL DESCRIPTION:
;  720          1632  1 !
;  721          1633  1 ! Output a double precision floating, the source is passed by value
;  722          1634  1 !
;  723          1635  1 ! FORMAL PARAMETERS:
;  724          1636  1 !
;  725          1637  1 !       ELEM.rd.v               double floating to print
;  726          1638  1 !
;  727          1639  1 ! IMPLICIT INPUTS:
;  728          1640  1 !
;  729          1641  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  730          1642  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  731          1643  1 !
;  732          1644  1 ! IMPLICIT OUTPUTS:
;  733          1645  1 !
;  734          1646  1 !       NONE
;  735          1647  1 !
;  736          1648  1 ! COMPLETION CODES:
;  737          1649  1 !
;  738          1650  1 !       NONE
;  739          1651  1 !
;  740          1652  1 ! SIDE EFFECTS:
;  741          1653  1 !
;  742          1654  1 !       NONE
;  743          1655  1 !
;  744          1656  1 !--
;  745          1657  1
;  746          1658  2     BEGIN
;  747          1659  2
;  748          1660  2     GLOBAL REGISTER
;  749          1661  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  750          1662  2
;  751          1663  2     CCB = .OTS$$A_CUR_LUB;
;  752          1664  2 !+
;  753          1665  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  754          1666  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  755          1667  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  756          1668  2 !-
;  757          1669  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_D,
;  758          1670  2                                                         ! data type
;  759          1671  2         K_DOUBLE_LEN,                                   ! length of data type
;  760          1672  2         ELEM,                                           ! address of value
;  761          1673  2         BAS$K_SEMI_FORM                                 ! semicolon format character
;  762          1674  2     );
;  763          1675  2     RETURN;
;  764          1676  1     END;                                                !End of BAS$OUT_D_V_S


                        0800 00000          .ENTRY  BAS$OUT_D_V_S, Save R11                         ; 1626
```

```
                    5B 00000000G   00   DO 00002          MOVL    OTS$$A_CUR_LUB, CCB                          ; 1663
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                                ; 1669
                    50 00000000G0040   DO 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0                  ;
                                   01   DD 00016          PUSHL   #1                                           ;
                                   04   AC   9F 00018     PUSHAB  ELEM                                         ;
                                   08   DD 0001B          PUSHL   #8                                           ;
                                   0B   DD 0001D          PUSHL   #11                                          ;
                    000000000G0040     04   FB 0001F     CALLS   #4, BAS$$AA_UDF_PR1[R0]                      ;
                                       04 00027          RET                                                  ; 1676

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 0173

;  765          1677  1
```

```
;  767          1678  1 GLOBAL ROUTINE BAS$OUT_D_V_C (                        ! print double by value, comma format
;  768          1679  1         ELEM                                          ! element to output by value
;  769          1680  1     ) : NOVALUE =                                     !
;  770          1681  1
;  771          1682  1 !++
;  772          1683  1 ! FUNCTIONAL DESCRIPTION:
;  773          1684  1 !
;  774          1685  1 ! Output a double precision floating, the source is passed by value
;  775          1686  1 !
;  776          1687  1 ! FORMAL PARAMETERS:
;  777          1688  1 !
;  778          1689  1 !       ELEM.rd.v               double floating to print
;  779          1690  1 !
;  780          1691  1 ! IMPLICIT INPUTS:
;  781          1692  1 !
;  782          1693  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  783          1694  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  784          1695  1 !
;  785          1696  1 ! IMPLICIT OUTPUTS:
;  786          1697  1 !
;  787          1698  1 !       NONE
;  788          1699  1 !
;  789          1700  1 ! COMPLETION CODES:
;  790          1701  1 !
;  791          1702  1 !       NONE
;  792          1703  1 !
;  793          1704  1 ! SIDE EFFECTS:
;  794          1705  1 !
;  795          1706  1 !       NONE
;  796          1707  1 !
;  797          1708  1 !--
;  798          1709  1
;  799          1710  2     BEGIN
;  800          1711  2
;  801          1712  2     GLOBAL REGISTER
;  802          1713  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  803          1714  2
;  804          1715  2     CCB = .OTS$$A_CUR_LUB;
;  805          1716  2 !+
;  806          1717  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  807          1718  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  808          1719  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  809          1720  2 !-
;  810          1721  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_D,
;  811          1722  2                                                         ! data type
;  812          1723  2         K_DOUBLE_LEN,                                   ! length of data type
;  813          1724  2         ELEM,                                           ! address of value
;  814          1725  2         BAS$K_COMMA_FOR                                 ! comma format character
;  815          1726  2     );
;  816          1727  2     RETURN;
;  817          1728  1     END;                                                !End of BAS$OUT_D_V_C


                        0800 00000          .ENTRY  BAS$OUT_D_V_C, Save R11                         ; 1678
```

```
                    5B 00000000G   00   D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ; 1715
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                           ; 1721
                    50 00000000G0040   D0 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0             :
                                   02   DD 00016          PUSHL   #2                                      :
                                04 AC   9F 00018          PUSHAB  ELEM                                    :
                                   08   DD 0001B          PUSHL   #8                                      :
                                   0B   DD 0001D          PUSHL   #11                                     :
                000000000G0040     04   FB 0001F          CALLS   #4, BAS$$AA_UDF_PR1[R0]                 :
                                        04 00027          RET                                             ; 1728

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 019B

;  818          1729  1
```

```
;  820          1730  1 GLOBAL ROUTINE BAS$OUT_D_V_B (                    ! print double by value, no format
;  821          1731  1         ELEM                                      ! element to output by value
;  822          1732  1     ) : NOVALUE =                                 !
;  823          1733  1
;  824          1734  1 !++
;  825          1735  1 ! FUNCTIONAL DESCRIPTION:
;  826          1736  1 !
;  827          1737  1 ! Output a double precision floating, the source is passed by value
;  828          1738  1 !
;  829          1739  1 ! FORMAL PARAMETERS:
;  830          1740  1 !
;  831          1741  1 !       ELEM.rd.v               double floating to print
;  832          1742  1 !
;  833          1743  1 ! IMPLICIT INPUTS:
;  834          1744  1 !
;  835          1745  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  836          1746  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  837          1747  1 !
;  838          1748  1 ! IMPLICIT OUTPUTS:
;  839          1749  1 !
;  840          1750  1 !       NONE
;  841          1751  1 !
;  842          1752  1 ! COMPLETION CODES:
;  843          1753  1 !
;  844          1754  1 !       NONE
;  845          1755  1 !
;  846          1756  1 ! SIDE EFFECTS:
;  847          1757  1 !
;  848          1758  1 !       NONE
;  849          1759  1 !
;  850          1760  1 !--
;  851          1761  1
;  852          1762  2     BEGIN
;  853          1763  2
;  854          1764  2     GLOBAL REGISTER
;  855          1765  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  856          1766  2
;  857          1767  2     CCB = .OTS$$A_CUR_LUB;
;  858          1768  2 !+
;  859          1769  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  860          1770  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  861          1771  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  862          1772  2 !-
;  863          1773  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_D,
;  864          1774  2                                                  ! data type
;  865          1775  2         K_DOUBLE_LEN,                            ! length of data type
;  866          1776  2         ELEM,                                    ! address of value
;  867          1777  2         BAS$K_NO_FORM                            ! no format character
;  868          1778  2     );
;  869          1779  2     RETURN;
;  870          1780  1     END;                                         !End of BAS$OUT_D_V_B


                        0800 00000          .ENTRY  BAS$OUT_D_V_B, Save R11                         ; 1730
```

```
                    5B 00000000G   00  D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ; 1767
                    50      FF71   CB  9A 00009          MOVZBL  -143(CCB), R0                           ; 1773
                    50 00000000G0040   D0 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                                   03  DD 00016          PUSHL   #3                                      ;
                                 04    AC  9F 00018      PUSHAB  ELEM                                    ;
                                       08  DD 0001B      PUSHL   #8                                      ;
                                       0B  DD 0001D      PUSHL   #11                                     ;
                 000000000G0040        04  FB 0001F      CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                           04 00027      RET                                             ; 1780

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 01C3

;  871          1781  1
```

```
;  873          1782  1 GLOBAL ROUTINE BAS$OUT_T_DX_S (                       ! print text by desc., semicolon format
;  874          1783  1         ELEM                                          ! element to output by descriptor
;  875          1784  1     ) : NOVALUE =                                     !
;  876          1785  1
;  877          1786  1 !++
;  878          1787  1 ! FUNCTIONAL DESCRIPTION:
;  879          1788  1 !
;  880          1789  1 ! Output text, the source is passed by descriptor
;  881          1790  1 !
;  882          1791  1 ! FORMAL PARAMETERS:
;  883          1792  1 !
;  884          1793  1 !       ELEM.rt.dx              text to print
;  885          1794  1 !
;  886          1795  1 ! IMPLICIT INPUTS:
;  887          1796  1 !
;  888          1797  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  889          1798  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  890          1799  1 !
;  891          1800  1 ! IMPLICIT OUTPUTS:
;  892          1801  1 !
;  893          1802  1 !       NONE
;  894          1803  1 !
;  895          1804  1 ! COMPLETION CODES:
;  896          1805  1 !
;  897          1806  1 !       NONE
;  898          1807  1 !
;  899          1808  1 ! SIDE EFFECTS:
;  900          1809  1 !
;  901          1810  1 !       NONE
;  902          1811  1 !
;  903          1812  1 !--
;  904          1813  1
;  905          1814  2     BEGIN
;  906          1815  2
;  907          1816  2     GLOBAL REGISTER
;  908          1817  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  909          1818  2
;  910          1819  2     MAP
;  911          1820  2         ELEM : REF BLOCK [8, BYTE];
;  912          1821  2
;  913          1822  2     CCB = .OTS$$A_CUR_LUB;
;  914          1823  2 !+
;  915          1824  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  916          1825  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  917          1826  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  918          1827  2 !-
;  919          1828  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_T,
;  920          1829  2                                                         ! data type
;  921          1830  2         .ELEM [DSC$W_LENGTH],                           ! length of data
;  922          1831  2         .ELEM,                                          ! address of desc
;  923          1832  2         BAS$K_SEMI_FORM                                 ! semicolon format character
;  924          1833  2     );
;  925          1834  2     RETURN;
;  926          1835  1     END;                                                !End of BAS$OUT_T_DX_S
```

```
                                    0800 00000          .ENTRY  BAS$OUT_T_DX_S, Save R11                        ; 1782
                          5B 00000000G  00    DO 00002  MOVL    OTS$$A_CUR_LOB, CCB                             ; 1822
                          50     FF71   CB    9A 00009  MOVZBL  -143(CCB), R0                                  ; 1828
                          50 00000000G0040   DO 0000E  MOVL    BAS$$AA_UDF_PR1-104[R0], R0                     ;
                                    01    DD 00016  PUSHL   #1                                                ;
                                 04   AC    DD 00018  PUSHL   ELEM                                            ; 1831
                             7E  04   BC    3C 0001B  MOVZWL  @ELEM, -(SP)                                    ; 1830
                                    0E    DD 0001F  PUSHL   #14                                               ; 1828
                    00000000G0040       04    FB 00021  CALLS   #4, BAS$$AA_UDF_PR1[R0]                       ;
                                        04 00029  RET                                                         ; 1835

; Routine Size:  42 bytes,    Routine Base:  _BAS$CODE + 01EB

;  927          1836  1
```

```
;  929          1837  1 GLOBAL ROUTINE BAS$OUT_T_DX_C (                      ! print text by desc., comma format
;  930          1838  1         ELEM                                         ! element to output by descriptor
;  931          1839  1     ) : NOVALUE =                                    !
;  932          1840  1
;  933          1841  1 !++
;  934          1842  1 ! FUNCTIONAL DESCRIPTION:
;  935          1843  1 !
;  936          1844  1 ! Output text, the source is passed by descriptor
;  937          1845  1 !
;  938          1846  1 ! FORMAL PARAMETERS:
;  939          1847  1 !
;  940          1848  1 !       ELEM.rt.dx              text to print
;  941          1849  1 !
;  942          1850  1 ! IMPLICIT INPUTS:
;  943          1851  1 !
;  944          1852  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  945          1853  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  946          1854  1 !
;  947          1855  1 ! IMPLICIT OUTPUTS:
;  948          1856  1 !
;  949          1857  1 !       NONE
;  950          1858  1 !
;  951          1859  1 ! COMPLETION CODES:
;  952          1860  1 !
;  953          1861  1 !       NONE
;  954          1862  1 !
;  955          1863  1 ! SIDE EFFECTS:
;  956          1864  1 !
;  957          1865  1 !       NONE
;  958          1866  1 !
;  959          1867  1 !--
;  960          1868  1
;  961          1869  2     BEGIN
;  962          1870  2
;  963          1871  2     GLOBAL REGISTER
;  964          1872  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  965          1873  2
;  966          1874  2     MAP
;  967          1875  2         ELEM : REF BLOCK [8, BYTE];
;  968          1876  2
;  969          1877  2     CCB = .OTS$$A_CUR_LUB;
;  970          1878  2 !+
;  971          1879  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  972          1880  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  973          1881  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  974          1882  2 !-
;  975          1883  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_T,
;  976          1884  2                                                     ! data type
;  977          1885  2         .ELEM [DSC$W_LENGTH],                       ! length of data
;  978          1886  2         .ELEM,                                      ! address of desc
;  979          1887  2         BAS$K_COMMA_FOR                             ! comma format character
;  980          1888  2     );
;  981          1889  2     RETURN;
;  982          1890  1     END;                                           !End of BAS$OUT_T_DX_C
```

```
                                    0800 00000          .ENTRY  BAS$OUT_T_DX_C, Save R11                        ; 1837
                          5B 00000000G  00    D0 00002  MOVL    OTS$$A_CUR_LUB, CCB                             ; 1877
                          50     FF71   CB    9A 00009  MOVZBL  -143(CCB), R0                                  ; 1883
                          50 00000000G0040    D0 0000E  MOVL    BAS$$AA_UDF_PR1-104[R0], R0                     ;
                                      02    DD 00016  PUSHL   #2                                              ;
                                04    AC    DD 00018  PUSHL   ELEM                                            ; 1886
                             7E       04    BC 3C 0001B  MOVZWL  @ELEM, -(SP)                                 ; 1885
                                      0E    DD 0001F  PUSHL   #14                                             ; 1883
                   000000000G0040             04    FB 00021  CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                                04 00029  RET                                                 ; 1890

; Routine Size:  42 bytes,    Routine Base:  _BAS$CODE + 0215

;  983          1891  1
```

```
;  985          1892  1 GLOBAL ROUTINE BAS$OUT_T_DX_B (                       ! print text by desc., no format
;  986          1893  1         ELEM                                          ! element to output by descriptor
;  987          1894  1     ) : NOVALUE =                                     !
;  988          1895  1
;  989          1896  1 !++
;  990          1897  1 ! FUNCTIONAL DESCRIPTION:
;  991          1898  1 !
;  992          1899  1 ! Output text, the source is passed by descriptor
;  993          1900  1 !
;  994          1901  1 ! FORMAL PARAMETERS:
;  995          1902  1 !
;  996          1903  1 !       ELEM.rt.dx              text to print
;  997          1904  1 !
;  998          1905  1 ! IMPLICIT INPUTS:
;  999          1906  1 !
; 1000          1907  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
; 1001          1908  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
; 1002          1909  1 !
; 1003          1910  1 ! IMPLICIT OUTPUTS:
; 1004          1911  1 !
; 1005          1912  1 !       NONE
; 1006          1913  1 !
; 1007          1914  1 ! COMPLETION CODES:
; 1008          1915  1 !
; 1009          1916  1 !       NONE
; 1010          1917  1 !
; 1011          1918  1 ! SIDE EFFECTS:
; 1012          1919  1 !
; 1013          1920  1 !       NONE
; 1014          1921  1 !
; 1015          1922  1 !--
; 1016          1923  1
; 1017          1924  2     BEGIN
; 1018          1925  2
; 1019          1926  2     GLOBAL REGISTER
; 1020          1927  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
; 1021          1928  2
; 1022          1929  2     MAP
; 1023          1930  2         ELEM : REF BLOCK [8, BYTE];
; 1024          1931  2
; 1025          1932  2     CCB = .OTS$$A_CUR_LUB;
; 1026          1933  2 !+
; 1027          1934  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
; 1028          1935  2 ! element transmit for prompting or for printing.  Therefore, based on the
; 1029          1936  2 ! statement type, either the INPUT or the PRINT UDF will be called.
; 1030          1937  2 !-
; 1031          1938  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_T,
; 1032          1939  2                                                         ! data type
; 1033          1940  2         .ELEM [DSC$W_LENGTH],                           ! length of data
; 1034          1941  2         .ELEM,                                          ! address of desc
; 1035          1942  2         BAS$K_NO_FORM                                   ! no format character
; 1036          1943  2     );
; 1037          1944  2     RETURN;
; 1038          1945  1     END;                                                !End of BAS$OUT_T_DX_B
```

```
                                 0800 00000     .ENTRY  BAS$OUT_T_DX_B, Save R11                  ; 1892
                5B 00000000G     00   D0 00002  MOVL    OTS$$A_CUR_LUB, CCB                      ; 1932
                50       FF71    CB   9A 00009  MOVZBL  -143(CCB), R0                           ; 1938
                50 00000000G0040 D0   D0 0000E  MOVL    BAS$$AA_UDF_PR1-104[R0], R0              ;
                                 03   DD 00016  PUSHL   #3                                       ;
                           04    AC   DD 00018  PUSHL   ELEM                                    ; 1941
                        7E 04    BC   3C 0001B  MOVZWL  @ELEM, -(SP)                            ; 1940
                                 0E   DD 0001F  PUSHL   #14                                     ; 1938
        000000000G0040           04   FB 00021  CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                 04   00029     RET                                            ; 1945

; Routine Size:  42 bytes,    Routine Base:  _BAS$CODE + 023F
```

```
;1040     1946  1 GLOBAL ROUTINE BAS$OUT_G_V_S (                    ! output G by value, semi format
;1041     1947  1         ELEM                                      ! element to output
;1042     1948  1     ) : NOVALUE =                                 !
;1043     1949  1
;1044     1950  1 !++
;1045     1951  1 ! FUNCTIONAL DESCRIPTION:
;1046     1952  1 !
;1047     1953  1 ! Output a g floating number passed by value, with a semi colon terminator
;1048     1954  1 !
;1049     1955  1 ! FORMAL PARAMETERS:
;1050     1956  1 !
;1051     1957  1 !       ELEM.rg.v               g float to print
;1052     1958  1 !
;1053     1959  1 ! IMPLICIT INPUTS:
;1054     1960  1 !
;1055     1961  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;1056     1962  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;1057     1963  1 !
;1058     1964  1 ! IMPLICIT OUTPUTS:
;1059     1965  1 !
;1060     1966  1 !       NONE
;1061     1967  1 !
;1062     1968  1 ! COMPLETION CODES:
;1063     1969  1 !
;1064     1970  1 !       NONE
;1065     1971  1 !
;1066     1972  1 ! SIDE EFFECTS:
;1067     1973  1 !
;1068     1974  1 !       NONE
;1069     1975  1 !
;1070     1976  1 !--
;1071     1977  1
;1072     1978  2     BEGIN
;1073     1979  2
;1074     1980  2     GLOBAL REGISTER
;1075     1981  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;1076     1982  2
;1077     1983  2     CCB = .OTS$$A_CUR_LUB;
;1078     1984  2 !+
;1079     1985  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;1080     1986  2 ! element transmit for prompting or for printing.  Therefore, based on the
;1081     1987  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;1082     1988  2 !-
;1083     1989  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_G,
;1084     1990  2                                                 ! data type
;1085     1991  2         K_GFLOAT_LEN,                           ! length of data type
;1086     1992  2         ELEM,                                   ! address of value
;1087     1993  2         BAS$K_SEMI_FORM                         ! semicolor format character
;1088     1994  2     );
;1089     1995  2     RETURN;
;1090     1996  1     END;                                        !End of BAS$OUT_G_V_S


                        0800 00000          .ENTRY  BAS$OUT_G_V_S, Save R11                         ; 1946
```

```
                5B 00000000G   00   DO 00002        MOVL    OTS$$A_CUR_LUB, CCB                     ; 1983
                50      FF71   CB   9A 00009        MOVZBL  -143(CCB), R0                           ; 1989
                50 00000000G0040   DO 0000E        MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                           01   DD 00016        PUSHL   #1                                      ;
                        04   AC   9F 00018        PUSHAB  ELEM                                    ;
                           08   DD 0001B        PUSHL   #8                                      ;
                           1B   DD 0001D        PUSHL   #27                                     ;
        00000000G0040              04   FB 0001F        CALLS   #4, BAS$$AA_UDF_PR1[R0]         ;
                                   04 00027        RET                                             ; 1996

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 0269
```

```
;  1092          1997  1 GLOBAL ROUTINE BAS$OUT_G_V_C (                        ! output g float by value, comma format
;  1093          1998  1         ELEM                                          ! element to print
;  1094          1999  1     ) : NOVALUE =                                     !
;  1095          2000  1
;  1096          2001  1 !++
;  1097          2002  1 ! FUNCTIONAL DESCRIPTION:
;  1098          2003  1 !
;  1099          2004  1 ! Print a gfloat number passed by value, with comma format
;  1100          2005  1 !
;  1101          2006  1 ! FORMAL PARAMETERS:
;  1102          2007  1 !
;  1103          2008  1 !       ELEM.rg.v               gfloat to print
;  1104          2009  1 !
;  1105          2010  1 ! IMPLICIT INPUTS:
;  1106          2011  1 !
;  1107          2012  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  1108          2013  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  1109          2014  1 !
;  1110          2015  1 ! IMPLICIT OUTPUTS:
;  1111          2016  1 !
;  1112          2017  1 !       NONE
;  1113          2018  1 !
;  1114          2019  1 ! COMPLETION CODES:
;  1115          2020  1 !
;  1116          2021  1 !       NONE
;  1117          2022  1 !
;  1118          2023  1 ! SIDE EFFECTS:
;  1119          2024  1 !
;  1120          2025  1 !       NONE
;  1121          2026  1 !
;  1122          2027  1 !--
;  1123          2028  1
;  1124          2029  2     BEGIN
;  1125          2030  2
;  1126          2031  2     GLOBAL REGISTER
;  1127          2032  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  1128          2033  2
;  1129          2034  2     CCB = .OTS$$A_CUR_LUB;
;  1130          2035  2 !+
;  1131          2036  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  1132          2037  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  1133          2038  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  1134          2039  2 !-
;  1135          2040  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_G,
;  1136          2041  2                                                         ! data type
;  1137          2042  2         K_GFLOAT_LEN,                                 ! length of data type
;  1138          2043  2         ELEM,                                         ! address of value
;  1139          2044  2         BAS$K_COMMA_FOR                               ! comma format character
;  1140          2045  2     );
;  1141          2046  2     RETURN;
;  1142          2047  1     END;                                              !End of BAS$OUT_G_V_C


                0800 00000          .ENTRY  BAS$OUT_G_V_C, Save R11                         ; 1997
```

```
                    5B 00000000G   00   D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ; 2034
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                           ; 2040
                    50 00000000G0040   D0 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                                   02   DD 00016          PUSHL   #2                                      ;
                                04     AC   9F 00018      PUSHAB  ELEM                                    ;
                                   08   DD 0001B          PUSHL   #8                                      ;
                                   1B   DD 0001D          PUSHL   #27                                     ;
                000000000G0040         04   FB 0001F      CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                         04 00027         RET                                             ; 2047

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 0291
```

```
;  1144          2048  1 GLOBAL ROUTINE BAS$OUT_G_V_B (                    ! print gfloat, blank format
;  1145          2049  1         ELEM                                      ! element to print
;  1146          2050  1     ) : NOVALUE =                                 !
;  1147          2051  1
;  1148          2052  1 !++
;  1149          2053  1 ! FUNCTIONAL DESCRIPTION:
;  1150          2054  1 !
;  1151          2055  1 ! Print a gfloat number passed by value, with a blank terminator
;  1152          2056  1 !
;  1153          2057  1 ! FORMAL PARAMETERS:
;  1154          2058  1 !
;  1155          2059  1 !       ELEM.rg.v               gfloat to print
;  1156          2060  1 !
;  1157          2061  1 ! IMPLICIT INPUTS:
;  1158          2062  1 !
;  1159          2063  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  1160          2064  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  1161          2065  1 !
;  1162          2066  1 ! IMPLICIT OUTPUTS:
;  1163          2067  1 !
;  1164          2068  1 !       NONE
;  1165          2069  1 !
;  1166          2070  1 ! COMPLETION CODES:
;  1167          2071  1 !
;  1168          2072  1 !       NONE
;  1169          2073  1 !
;  1170          2074  1 ! SIDE EFFECTS:
;  1171          2075  1 !
;  1172          2076  1 !       NONE
;  1173          2077  1 !
;  1174          2078  1 !--
;  1175          2079  1
;  1176          2080  2     BEGIN
;  1177          2081  2
;  1178          2082  2     GLOBAL REGISTER
;  1179          2083  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  1180          2084  2
;  1181          2085  2     CCB = .OTS$$A_CUR_LUB;
;  1182          2086  2 !+
;  1183          2087  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  1184          2088  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  1185          2089  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  1186          2090  2 !-
;  1187          2091  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_G,
;  1188          2092  2                                                  ! data type
;  1189          2093  2         K_GFLOAT_LEN,                            ! length of data type
;  1190          2094  2         ELEM,                                    ! address of value
;  1191          2095  2         BAS$K_NO_FORM                            ! no format character
;  1192          2096  2     );
;  1193          2097  2     RETURN;
;  1194          2098  1     END;                                         !End of BAS$OUT_G_V_B


                    0800 00000         .ENTRY  BAS$OUT_G_V_B, Save R11                    ; 2048
```

```
                    5B 00000000G   00   DO 00002          MOVL    OTS$$A_CUR_LUB, CCB                          ; 2085
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                                ; 2091
                    50 00000000G0040   DO 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0                  :
                                   03   DD 00016          PUSHL   #3                                           :
                                04 AC   9F 00018          PUSHAB  ELEM                                         :
                                   08   DD 0001B          PUSHL   #8                                           :
                                   1B   DD 0001D          PUSHL   #27                                          :
                00000000G0040          04   FB 0001F     CALLS   #4, BAS$$AA_UDF_PR1[R0]                      :
                                            04 00027     RET                                                  ; 2098

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 02B9
```

```
;1196          2099  1 GLOBAL ROUTINE BAS$IN_B_R (                          ! input byte by ref
;1197          2100  1         ELEM                                         ! element to input by reference
;1198          2101  1     ) : NOVALUE =                                    !
;1199          2102  1
;1200          2103  1 !++
;1201          2104  1 ! FUNCTIONAL DESCRIPTION:
;1202          2105  1 !
;1203          2106  1 ! Input a byte, the destination is passed by reference
;1204          2107  1 !
;1205          2108  1 ! FORMAL PARAMETERS:
;1206          2109  1 !
;1207          2110  1 !       ELEM.rb.r               where to store the byte input
;1208          2111  1 !
;1209          2112  1 ! IMPLICIT INPUTS:
;1210          2113  1 !
;1211          2114  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;1212          2115  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;1213          2116  1 !
;1214          2117  1 ! IMPLICIT OUTPUTS:
;1215          2118  1 !
;1216          2119  1 !       NONE
;1217          2120  1 !
;1218          2121  1 ! COMPLETION CODES:
;1219          2122  1 !
;1220          2123  1 !       NONE
;1221          2124  1 !
;1222          2125  1 ! SIDE EFFECTS:
;1223          2126  1 !
;1224          2127  1 !       NONE
;1225          2128  1 !
;1226          2129  1 !--
;1227          2130  1
;1228          2131  2     BEGIN
;1229          2132  2
;1230          2133  2     GLOBAL REGISTER
;1231          2134  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;1232          2135  2
;1233          2136  2     CCB = .OTS$$A_CUR_LUB;
;1234          2137  2     BAS$$UDF_RL1 (DSC$K_DTYPE_B,                       ! data type byte
;1235          2138  2         K_GFLOAT_LEN,                                  ! length of data type
;1236          2139  2         .ELEM,                                         ! address of destination
;1237          2140  2         BAS$K_NULL);                                   ! null format character
;1238          2141  2     RETURN;
;1239          2142  1     END;                                               !End of BAS$IN_B_R


                                        0800 00000          .ENTRY  BAS$IN_B_R, Save R11                    ;2099
                5B 00000000G    00      D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ;2136
                                7E      D4 00009          CLRL    -(SP)                                   ;2137
                          04    AC      DD 0000B          PUSHL   ELEM                                    ;2139
                                08      DD 0000E          PUSHL   #8                                      ;2137
                                06      DD 00010          PUSHL   #6                                      ;
      00000000G   00                    04 FB 00012       CALLS   #4, BAS$$UDF_RL1                        ;
                                        04 00019          RET                                             ;2142
```

; Routine Size: 26 bytes,   Routine Base: _BAS$CODE + 02E1

```
;  1241          2143  1 GLOBAL ROUTINE BAS$OUT_H_V_S (                    ! print hfloat with semi format
;  1242          2144  1         ELEM                                      ! element to print
;  1243          2145  1     ) : NOVALUE =                                 !
;  1244          2146  1
;  1245          2147  1 !++
;  1246          2148  1 ! FUNCTIONAL DESCRIPTION:
;  1247          2149  1 !
;  1248          2150  1 ! Print an hfloat number passed by value, with semi colon format
;  1249          2151  1 !
;  1250          2152  1 ! FORMAL PARAMETERS:
;  1251          2153  1 !
;  1252          2154  1 !       ELEM.rh.v               hfloat to print
;  1253          2155  1 !
;  1254          2156  1 ! IMPLICIT INPUTS:
;  1255          2157  1 !
;  1256          2158  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  1257          2159  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  1258          2160  1 !
;  1259          2161  1 ! IMPLICIT OUTPUTS:
;  1260          2162  1 !
;  1261          2163  1 !       NONE
;  1262          2164  1 !
;  1263          2165  1 ! COMPLETION CODES:
;  1264          2166  1 !
;  1265          2167  1 !       NONE
;  1266          2168  1 !
;  1267          2169  1 ! SIDE EFFECTS:
;  1268          2170  1 !
;  1269          2171  1 !       NONE
;  1270          2172  1 !
;  1271          2173  1 !--
;  1272          2174  1
;  1273          2175  2     BEGIN
;  1274          2176  2
;  1275          2177  2     GLOBAL REGISTER
;  1276          2178  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  1277          2179  2
;  1278          2180  2     CCB = .OTS$$A_CUR_LUB;
;  1279          2181  2 !+
;  1280          2182  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  1281          2183  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  1282          2184  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  1283          2185  2 !-
;  1284          2186  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_H,
;  1285          2187  2                                                   ! data type
;  1286          2188  2         K_HFLOAT_LEN,                             ! length of data type
;  1287          2189  2         ELEM,                                     ! address of value
;  1288          2190  2         BAS$K_SEMI_FORM                           ! semicolon format character
;  1289          2191  2     );
;  1290          2192  2     RETURN;
;  1291          2193  1     END;                                          !End of BAS$OUT_H_V_S


                                0800 00000          .ENTRY  BAS$OUT_H_V_S, Save R11                         ; 2143
```

```
                    5B 00000000G   00   D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                          ; 2180
                    50      FF71   CB   9A 00009          MOVZBL  -143(CCB), R0                                ; 2186
                    50 00000000G0040   D0 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0                  ;
                                   01   DD 00016          PUSHL   #1                                           ;
                                   04   AC   9F 00018     PUSHAB  ELEM                                         ;
                                   10   DD 0001B          PUSHL   #16                                          ;
                                   1C   DD 0001D          PUSHL   #28                                          ;
                        00000000G0040   04   FB 0001F     CALLS   #4, BAS$$AA_UDF_PR1[R0]                      ;
                                        04 00027          RET                                                  ; 2193

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 02FB
```

```
;  1293          2194  1 GLOBAL RO  INE BAS$OUT_H_V_C (                    ! print hfloat with comma format
;  1294          2195  1         EL  M                                     ! element to print
;  1295          2196  1     ) : NO  ALUE =                                !
;  1296          2197  1
;  1297          2198  1 !++
;  1298          2199  1 ! FUNCTIONAL DESCRIPTION:
;  1299          2200  1 !
;  1300          2201  1 ! Print an hfloat number passed by value, with comma format
;  1301          2202  1 !
;  1302          2203  1 ! FORMAL PARAMETERS:
;  1303          2204  1 !
;  1304          2205  1 !       ELEM.rh.v               hfloat to print
;  1305          2206  1 !
;  1306          2207  1 ! IMPLICIT INPUTS:
;  1307          2208  1 !
;  1308          2209  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  1309          2210  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  1310          2211  1 !
;  1311          2212  1 ! IMPLICIT OUTPUTS:
;  1312          2213  1 !
;  1313          2214  1 !       NONE
;  1314          2215  1 !
;  1315          2216  1 ! COMPLETION CODES:
;  1316          2217  1 !
;  1317          2218  1 !       NONE
;  1318          2219  1 !
;  1319          2220  1 ! SIDE EFFECTS:
;  1320          2221  1 !
;  1321          2222  1 !       NONE
;  1322          2223  1 !
;  1323          2224  1 !--
;  1324          2225  1
;  1325          2226  2     BEGIN
;  1326          2227  2
;  1327          2228  2     GLOBAL REGISTER
;  1328          2229  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  1329          2230  2
;  1330          2231  2     CCB = .OTS$$A_CUR_LUB;
;  1331          2232  2 !+
;  1332          2233  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  1333          2234  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  1334          2235  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  1335          2236  2 !-
;  1336          2237  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_H,
;  1337          2238  2                                                 ! data type
;  1338          2239  2         K_HFLOAT_LEN,                           ! length of data type
;  1339          2240  2         ELEM,                                   ! address of value
;  1340          2241  2         BAS$K_COMMA_FOR                         ! comma format character
;  1341          2242  2     );
;  1342          2243  2     RETURN;
;  1343          2244  1     END;                                        !End of BAS$OUT_H_DX_C


                                0800 00000          .ENTRY  BAS$OUT_H_V_C, Save R11                         ; 2194
```

```
                    5B 00000000G   00   D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ; 2231
                    50     FF71    CB   9A 00009          MOVZBL  -143(CCB), R0                           ; 2237
                    50 00000000G0040   D0 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0             ;
                                   02   DD 00016          PUSHL   #2                                      ;
                                04     AC   9F 00018      PUSHAB  ELEM                                    ;
                                   10   DD 0001B          PUSHL   #16                                     ;
                                   1C   DD 0001D          PUSHL   #28                                     ;
                000000000G0040     04   FB 0001F          CALLS   #4, BAS$$AA_UDF_PR1[R0]                 ;
                                        04 00027          RET                                             ; 2244

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 0323
```

```
;  1345          2245  1 GLOBAL ROUTINE BAS$OUT_H_V_B (                        ! print hfloat with blank format
;  1346          2246  1         ELEM                                          ! element to print
;  1347          2247  1     ) : NOVALUE =                                     !
;  1348          2248  1
;  1349          2249  1 !++
;  1350          2250  1 ! FUNCTIONAL DESCRIPTION:
;  1351          2251  1 !
;  1352          2252  1 ! Print an hfloat number passed by value, with blank format
;  1353          2253  1 !
;  1354          2254  1 ! FORMAL PARAMETERS:
;  1355          2255  1 !
;  1356          2256  1 !       ELEM.rh.v               hfloat to print
;  1357          2257  1 !
;  1358          2258  1 ! IMPLICIT INPUTS:
;  1359          2259  1 !
;  1360          2260  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  1361          2261  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  1362          2262  1 !
;  1363          2263  1 ! IMPLICIT OUTPUTS:
;  1364          2264  1 !
;  1365          2265  1 !       NONE
;  1366          2266  1 !
;  1367          2267  1 ! COMPLETION CODES:
;  1368          2268  1 !
;  1369          2269  1 !       NONE
;  1370          2270  1 !
;  1371          2271  1 ! SIDE EFFECTS:
;  1372          2272  1 !
;  1373          2273  1 !       NONE
;  1374          2274  1 !
;  1375          2275  1 !--
;  1376          2276  1
;  1377          2277  2     BEGIN
;  1378          2278  2
;  1379          2279  2     GLOBAL REGISTER
;  1380          2280  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  1381          2281  2
;  1382          2282  2     CCB = .OTS$$A_CUR_LUB;
;  1383          2283  2 !+
;  1384          2284  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  1385          2285  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  1386          2286  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  1387          2287  2 !-
;  1388          2288  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_H,
;  1389          2289  2                                                         ! data type
;  1390          2290  2         K_HFLOAT_LEN,                                 ! length of data type
;  1391          2291  2         ELEM,                                         ! address of value
;  1392          2292  2         BAS$K_NO_FORM                                 ! no format character
;  1393          2293  2     );
;  1394          2294  2     RETURN;
;  1395          2295  1     END;                                              !End of BAS$OUT_H_DX_B


                              0800 00000         .ENTRY  BAS$OUT_H_V_B, Save R11                         ; 2245
```

```
                          5B 00000000G   00    DO 00002          MOVL    OTS$$A_CUR_LUB, CCB                          ; 2282
                          50      FF71   CB    9A 00009          MOVZBL  -143(CCB), R0                                ; 2288
                          50 00000000G0040     DO 0000E          MOVL    BAS$$AA_UDF_PR1-104[R0], R0                  ;
                                         03    DD 00016          PUSHL   #3                                           ;
                                   04    AC    9F 00018          PUSHAB  ELEM                                         ;
                                         10    DD 0001B          PUSHL   #16                                          ;
                                         1C    DD 0001D          PUSHL   #28                                          ;
                    00000000G0040              04    FB 0001F    CALLS   #4, BAS$$AA_UDF_PR1[R0]                      ;
                                               04 00027          RET                                                  ; 2295

; Routine Size:  40 bytes,    Routine Base:  _BAS$CODE + 034B
```

```
;1397     2296  1 GLOBAL ROUTINE BAS$OUT_P_DX_S (                      ! print packed with semi format
;1398     2297  1         ELEM                                         ! element to print
;1399     2298  1     ) : NOVALUE =                                    !
;1400     2299  1
;1401     2300  1 !++
;1402     2301  1 ! FUNCTIONAL DESCRIPTION:
;1403     2302  1 !
;1404     2303  1 ! Print a packed decimal value passed by desc, with semi colon format
;1405     2304  1 !
;1406     2305  1 ! FORMAL PARAMETERS:
;1407     2306  1 !
;1408     2307  1 !       ELEM.rp.dx              packed to print
;1409     2308  1 !
;1410     2309  1 ! IMPLICIT INPUTS:
;1411     2310  1 !
;1412     2311  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;1413     2312  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;1414     2313  1 !
;1415     2314  1 ! IMPLICIT OUTPUTS:
;1416     2315  1 !
;1417     2316  1 !       NONE
;1418     2317  1 !
;1419     2318  1 ! COMPLETION CODES:
;1420     2319  1 !
;1421     2320  1 !       NONE
;1422     2321  1 !
;1423     2322  1 ! SIDE EFFECTS:
;1424     2323  1 !
;1425     2324  1 !       NONE
;1426     2325  1 !
;1427     2326  1 !--
;1428     2327  1
;1429     2328  2     BEGIN
;1430     2329  2
;1431     2330  2     GLOBAL REGISTER
;1432     2331  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;1433     2332  2
;1434     2333  2     MAP
;1435     2334  2         ELEM : REF BLOCK [8,BYTE];
;1436     2335  2
;1437     2336  2     CCB = .OTS$$A_CUR_LUB;
;1438     2337  2 !+
;1439     2338  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;1440     2339  2 ! element transmit for prompting or for printing.  Therefore, based on the
;1441     2340  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;1442     2341  2 !-
;1443     2342  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_P,
;1444     2343  2                                                         ! data type
;1445     2344  2         .ELEM [DSC$W_LENGTH],                           ! length of data type
;1446     2345  2         .ELEM,                                          ! address of value
;1447     2346  2         BAS$K_SEMI_FORM                                 ! semicolon format character
;1448     2347  2     );
;1449     2348  2     RETURN;
;1450     2349  1     END;                                                !End of BAS$OUT_B_V_S
```

```
                        0800 00000      .ENTRY  BAS$OUT_P_DX_S, Save R11                   ; 2296
        5B 00000000G  00  D0 00002      MOVL    OTS$$A_CUR_LUB, CCB                        ; 2336
        50      FF71  CB  9A 00009      MOVZBL  -143(CCB), R0                              ; 2342
        50 00000000G0040  D0 0000E      MOVL    BAS$$AA_UDF_PR1-104[R0], R0                ;
                          01  DD 00016  PUSHL   #1                                         ;
                    04  AC  DD 00018    PUSHL   ELEM                                       ; 2345
        7E          04  BC  3C 0001B    MOVZWL  @ELEM, -(SP)                               ; 2344
                          15  DD 0001F  PUSHL   #21                                        ; 2342
 00000000G0040            04  FB 00021  CALLS   #4, BAS$$AA_UDF_PR1[R0]                    ;
                              04 00029  RET                                                ; 2349

; Routine Size:  42 bytes,    Routine Base:  _BAS$CODE + 0373
```

```
;  1452          2350  1 GLOBAL ROUTINE BAS$OUT_P_DX_C (                      ! print packed with comma format
;  1453          2351  1         ELEM                                         ! element to print
;  1454          2352  1     ) : NOVALUE =                                    !
;  1455          2353  1
;  1456          2354  1 !++
;  1457          2355  1 ! FUNCTIONAL DESCRIPTION:
;  1458          2356  1 !
;  1459          2357  1 ! Print a packed decimal value passed by desc, with comma format
;  1460          2358  1 !
;  1461          2359  1 ! FORMAL PARAMETERS:
;  1462          2360  1 !
;  1463          2361  1 !       ELEM.rp.dx              packed to print
;  1464          2362  1 !
;  1465          2363  1 ! IMPLICIT INPUTS:
;  1466          2364  1 !
;  1467          2365  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  1468          2366  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  1469          2367  1 !
;  1470          2368  1 ! IMPLICIT OUTPUTS:
;  1471          2369  1 !
;  1472          2370  1 !       NONE
;  1473          2371  1 !
;  1474          2372  1 ! COMPLETION CODES:
;  1475          2373  1 !
;  1476          2374  1 !       NONE
;  1477          2375  1 !
;  1478          2376  1 ! SIDE EFFECTS:
;  1479          2377  1 !
;  1480          2378  1 !       NONE
;  1481          2379  1 !
;  1482          2380  1 !--
;  1483          2381  1
;  1484          2382  2     BEGIN
;  1485          2383  2
;  1486          2384  2     GLOBAL REGISTER
;  1487          2385  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  1488          2386  2
;  1489          2387  2     MAP
;  1490          2388  2         ELEM : REF BLOCK [8, BYTE];
;  1491          2389  2
;  1492          2390  2     CCB = .OTS$$A_CUR_LUB;
;  1493          2391  2 !+
;  1494          2392  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  1495          2393  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  1496          2394  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  1497          2395  2 !-
;  1498          2396  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_P,
;  1499          2397  2                                                     ! data type
;  1500          2398  2         .ELEM [DSC$W_LENGTH],                       ! length of data type
;  1501          2399  2         .ELEM,                                      ! address of value
;  1502          2400  2         BAS$K_COMMA_FOR                             ! comma format character
;  1503          2401  2     );
;  1504          2402  2     RETURN;
;  1505          2403  1     END;                                            !End of BAS$OUT_P_DX_C
```

```
                                    0800 00000          .ENTRY  BAS$OUT_P_DX_C, Save R11                        ; 2350
                          5B 00000000G  00   D0 00002   MOVL    OTS$$A_CUR_LUB, CCB                             ; 2390
                          50     FF71   CB   9A 00009   MOVZBL  -143(CCB), R0                                   ; 2396
                          50 00000000G0040  D0 0000E   MOVL    BAS$$AA_UDF_PR1-104[R0], R0                     ;
                                    02   DD 00016   PUSHL   #2                                                ;
                                 04      AC   DD 00018   PUSHL   ELEM                                          ; 2399
                             7E  04      BC   3C 0001B   MOVZWL  @ELEM, -(SP)                                  ; 2398
                                    15   DD 0001F   PUSHL   #21                                               ; 2396
                  00000000G0040         04   FB 00021   CALLS   #4, BAS$$AA_UDF_PR1[R0]                       ;
                                        04   04 00029   RET                                                    ; 2403

; Routine Size:  42 bytes,    Routine Base:  _BAS$CODE + 039D
```

```
;  1507          2404  1 GLOBAL ROUTINE BAS$OUT_P_DX_B(                        ! print packed with blank format
;  1508          2405  1         ELEM                                          ! element to print
;  1509          2406  1     ) : NOVALUE =                                     !
;  1510          2407  1
;  1511          2408  1 !++
;  1512          2409  1 ! FUNCTIONAL DESCRIPTION:
;  1513          2410  1 !
;  1514          2411  1 ! Print a packed decimal value passed by desc, with blank format
;  1515          2412  1 !
;  1516          2413  1 ! FORMAL PARAMETERS:
;  1517          2414  1 !
;  1518          2415  1 !       ELEM.rp.dx              packed to print
;  1519          2416  1 !
;  1520          2417  1 ! IMPLICIT INPUTS:
;  1521          2418  1 !
;  1522          2419  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;  1523          2420  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;  1524          2421  1 !
;  1525          2422  1 ! IMPLICIT OUTPUTS:
;  1526          2423  1 !
;  1527          2424  1 !       NONE
;  1528          2425  1 !
;  1529          2426  1 ! COMPLETION CODES:
;  1530          2427  1 !
;  1531          2428  1 !       NONE
;  1532          2429  1 !
;  1533          2430  1 ! SIDE EFFECTS:
;  1534          2431  1 !
;  1535          2432  1 !       NONE
;  1536          2433  1 !
;  1537          2434  1 !--
;  1538          2435  1
;  1539          2436  2     BEGIN
;  1540          2437  2
;  1541          2438  2     GLOBAL REGISTER
;  1542          2439  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;  1543          2440  2
;  1544          2441  2     MAP
;  1545          2442  2         ELEM : REF BLOCK [8, BYTE];
;  1546          2443  2
;  1547          2444  2     CCB = .OTS$$A_CUR_LUB;
;  1548          2445  2 !+
;  1549          2446  2 ! Dispatch to the UDF level.  Dispatching is done because this may be a PRINT
;  1550          2447  2 ! element transmit for prompting or for printing.  Therefore, based on the
;  1551          2448  2 ! statement type, either the INPUT or the PRINT UDF will be called.
;  1552          2449  2 !-
;  1553          2450  2     (BAS$$AA_UDF_PR1 + .BAS$$AA_UDF_PR1 [.CCB [ISB$B_STTM_TYPE] - ISB$K_BASSTTYLO + 1]) (DSC$K_DTYPE_P,
;  1554          2451  2                                                         ! data type
;  1555          2452  2         .ELEM [DSC$W_LENGTH],                           ! length of data type
;  1556          2453  2         .ELEM,                                          ! address of value
;  1557          2454  2         BAS$K_NO_FORM                                   ! no format character
;  1558          2455  2     );
;  1559          2456  2     RETURN;
;  1560          2457  1     END;                                                !End of BAS$OUT_P_DX_B
```

```
                                    0800 00000          .ENTRY  BAS$OUT_P_DX_B, Save R11                        ; 2404
                          5B 00000000G  00   D0 00002   MOVL    OTS$$A_CUR_LUB, CCB                             ; 2444
                          50     FF71   CB   9A 00009   MOVZBL  -143(CCB), R0                                   ; 2450
                          50 00000000G0040   D0 0000E   MOVL    BAS$$AA_UDF_PR1-104[R0], R0                     ;
                                         03   DD 00016   PUSHL   #3                                            ;
                                   04    AC   DD 00018   PUSHL   ELEM                                          ; 2453
                             7E    04    BC   3C 0001B   MOVZWL  @ELEM, -(SP)                                  ; 2452
                                         15   DD 0001F   PUSHL   #21                                          ; 2450
                   000000000G0040        04   FB 00021   CALLS   #4, BAS$$AA_UDF_PR1[R0]                      ;
                                              04 00029   RET                                                   ; 2457

; Routine Size:  42 bytes,    Routine Base:  _BAS$CODE + 03C7
```

```
;1562     2458  1 GLOBAL ROUTINE BAS$IN_G_R (                          ! input gfloat by ref
;1563     2459  1         ELEM                                         ! element to input by reference
;1564     2460  1     ) : NOVALUE =                                    !
;1565     2461  1
;1566     2462  1 !++
;1567     2463  1 ! FUNCTIONAL DESCRIPTION:
;1568     2464  1 !
;1569     2465  1 ! Input a gfloat, the destination is passed by reference
;1570     2466  1 !
;1571     2467  1 ! FORMAL PARAMETERS:
;1572     2468  1 !
;1573     2469  1 !       ELEM.rg.r               where to store the gfloat input
;1574     2470  1 !
;1575     2471  1 ! IMPLICIT INPUTS:
;1576     2472  1 !
;1577     2473  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;1578     2474  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;1579     2475  1 !
;1580     2476  1 ! IMPLICIT OUTPUTS:
;1581     2477  1 !
;1582     2478  1 !       NONE
;1583     2479  1 !
;1584     2480  1 ! COMPLETION CODES:
;1585     2481  1 !
;1586     2482  1 !       NONE
;1587     2483  1 !
;1588     2484  1 ! SIDE EFFECTS:
;1589     2485  1 !
;1590     2486  1 !       NONE
;1591     2487  1 !
;1592     2488  1 !--
;1593     2489  1
;1594     2490  2     BEGIN
;1595     2491  2
;1596     2492  2     GLOBAL REGISTER
;1597     2493  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;1598     2494  2
;1599     2495  2     CCB = .OTS$$A_CUR_LUB;
;1600     2496  2     BAS$$UDF_RL1 (DSC$K_DTYPE_G,                       ! data type gfloat
;1601     2497  2         K_GFLOAT_LEN,                                  ! length of data type
;1602     2498  2         .ELEM,                                         ! address of destination
;1603     2499  2         BAS$K_NULL);                                   ! null format character
;1604     2500  2     RETURN;
;1605     2501  1     END;                                               !End of BAS$IN_G_R


                                    0800 00000         .ENTRY  BAS$IN_G_R, Save R11                    ;2458
                    5B 00000000G    00   D0 00002     MOVL    OTS$$A_CUR_LUB, CCB                     ;2495
                                    7E   D4 00009     CLRL    -(SP)                                   ;2496
                               04   AC   DD 0000B     PUSHL   ELEM                                    ;2498
                                    08   DD 0000E     PUSHL   #8                                      ;2496
                                    1B   DD 00010     PUSHL   #27                                     ;
                    00000000G   00  04   FB 00012     CALLS   #4, BAS$$UDF_RL1                        ;
                                        04 00019     RET                                             ;2501
```

; Routine Size: 26 bytes,   Routine Base: _BAS$CODE + 03F1

```
;1607     2502  1 GLOBAL ROUTINE BAS$IN_H_R (                    ! input hfloat by ref
;1608     2503  1         ELEM                                   ! element to input by reference
;1609     2504  1     ) : NOVALUE =                              !
;1610     2505  1
;1611     2506  1 !++
;1612     2507  1 ! FUNCTIONAL DESCRIPTION:
;1613     2508  1 !
;1614     2509  1 ! Input an hfloat, the destination is passed by reference
;1615     2510  1 !
;1616     2511  1 ! FORMAL PARAMETERS:
;1617     2512  1 !
;1618     2513  1 !       ELEM.rh.r               where to store the hfloat input
;1619     2514  1 !
;1620     2515  1 ! IMPLICIT INPUTS:
;1621     2516  1 !
;1622     2517  1 !       OTS$$A_CUR_LUB          current Logical Unit Block
;1623     2518  1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
;1624     2519  1 !
;1625     2520  1 ! IMPLICIT OUTPUTS:
;1626     2521  1 !
;1627     2522  1 !       NONE
;1628     2523  1 !
;1629     2524  1 ! COMPLETION CODES:
;1630     2525  1 !
;1631     2526  1 !       NONE
;1632     2527  1 !
;1633     2528  1 ! SIDE EFFECTS:
;1634     2529  1 !
;1635     2530  1 !       NONE
;1636     2531  1 !
;1637     2532  1 !--
;1638     2533  1
;1639     2534  2     BEGIN
;1640     2535  2
;1641     2536  2     GLOBAL REGISTER
;1642     2537  2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
;1643     2538  2
;1644     2539  2     CCB = .OTS$$A_CUR_LUB;
;1645     2540  2     BAS$$UDF_RL1 (DSC$K_DTYPE_H,                ! data type hfloat
;1646     2541  2         K_HFLOAT_LEN,                           ! length of data type
;1647     2542  2         .ELEM,                                  ! address of destination
;1648     2543  2         BAS$K_NULL);                            ! null format character
;1649     2544  2     RETURN;
;1650     2545  1     END;                                        !End of BAS$IN_H_R


                                    0800 00000          .ENTRY  BAS$IN_H_R, Save R11                    ;2502
                5B 00000000G    00  D0 00002          MOVL    OTS$$A_CUR_LUB, CCB                     ;2539
                                7E  D4 00009          CLRL    -(SP)                                   ;2540
                          04    AC  DD 0000B          PUSHL   ELEM                                    ;2542
                                10  DD 0000E          PUSHL   #16                                     ;2540
                                1C  DD 00010          PUSHL   #28                                     ;
                00000000G   00      04  FB 00012     CALLS   #4, BAS$$UDF_RL1                        ;
                                        04 00019      RET                                             ;2545
```

```
; Routine Size:  26 bytes,    Routine Base:  _BAS$CODE + 040B

; 1651          2546  1
```

```
1 GLOBAL ROUTINE BAS$IN_P_DX (                              ! input packed decimal by desc.
1         ELEM                                              ! element to input by reference
1     ) : NOVALUE =                                         !
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 ! Input packed decimal, the destination is passed by descriptor
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       ELEM.rp.dx              where to store the packed input
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       OTS$$A_CUR_LUB          current Logical Unit Block
1 !       ISB$B_STTM_TYPE         statement type of this I/O statement
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! COMPLETION CODES:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       NONE
1 !
1 !--
1
2     BEGIN
2
2     GLOBAL REGISTER
2         CCB = K_CCB_REG : REF BLOCK [, BYTE];
2
2     MAP
2         ELEM : REF BLOCK [8, BYTE];
2
2     CCB = .OTS$$A_CUR_LUB;
2     BAS$$UDF_RL1 (DSC$K_DTYPE_P,                          ! data type
2         ELEM [DSC$W_LENGTH],                              ! string length
2         .ELEM,                                            ! address of descriptor
2         BAS$K_NULL);                                      ! null format character
2     RETURN;
1     END;                                                  !End of BAS$IN_P_DX
```

```
                    0800 00000      .ENTRY  BAS$IN_P_DX, Save R11        ; 2547
5B 00000000G   00     D0 00002      MOVL    OTS$$A_CUR_LUB, CCB          ; 2587
               7E     D4 00009      CLRL    -(SP)                        ; 2589
       04      AC     DD 0000B      PUSHL   ELEM                         ; 2590
       04      AC     DD 0000E      PUSHL   ELEM                         ; 2589
```

```
                                 15  DD 00011         PUSHL   #21                          ;
                 00000000G  00   04  FB 00013         CALLS   #4, BAS$$UDF_RL1             ;
                                     04 0001A         RET                                  ; 2593

; Routine Size:  27 bytes,    Routine Base:  _BAS$CODE + 0425


; 1700          2594  1
; 1701          2595  1
```

```
;  1 END                                        !End of module - BAS$UPI_TERM_IO
;  1
;  0 ELUDOM
```

```
;                                  PSECT SUMMARY
;
;      Name                      Bytes                    Attributes
;
;  _BAS$CODE                       1088  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       10         0        581        00:01.2
```

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

; Size:          1088 code + 0 data bytes
; Run Time:         00:27.5
; Elapsed Time:     01:10.3
; Lines/CPU Min:      5676
; Lexemes/CPU-Min: 22450
; Memory Used:  130 pages
; Compilation Complete
```

BASUDFWF
LIS

BASSTRING
LIS

BASTERMIO
LIS

BASTRM
LIS

BASTAB
LIS

BASUDFRM
LIS

BASSYS
LIS

BASSTR
LIS

BASUDFRL
LIS