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

FILEID**BASUDFRM

```
BBBBBBBB      AAAAAA      SSSSSSSS  UU      UU  DDDDDDDD    FFFFFFFFFF  RRRRRRRR    MM       MM
BBBBBBBB      AAAAAA      SSSSSSSS  UU      UU  DDDDDDDD    FFFFFFFFFF  RRRRRRRR    MM       MM
BB      BB  AA      AA  SS        UU      UU  DD      DD  FF          RR      RR  MMMM   MMMM
BB      BB  AA      AA  SS        UU      UU  DD      DD  FF          RR      RR  MMMM   MMMM
BB      BB  AA      AA  SS        UU      UU  DD      DD  FF          RR      RR  MM  MM  MM
BB      BB  AA      AA  SS        UU      UU  DD      DD  FF          RR      RR  MM  MM  MM
BBBBBBBB    AA      AA    SSSSSS  UU      UU  DD      DD  FFFFFFFF    RRRRRRRR    MM       MM
BBBBBBBB    AA      AA    SSSSSS  UU      UU  DD      DD  FFFFFFFF    RRRRRRRR    MM       MM
BB      BB  AAAAAAAAAA          SS  UU      UU  DD      DD  FF          RR  RR      MM       MM
BB      BB  AAAAAAAAAA          SS  UU      UU  DD      DD  FF          RR  RR      MM       MM
BB      BB  AA      AA          SS  UU      UU  DD      DD  FF          RR    RR    MM       MM      ....
BB      BB  AA      AA          SS  UU      UU  DD      DD  FF          RR    RR    MM       MM      ....
BBBBBBBB    AA      AA  SSSSSSSS    UUUUUUUUUU  DDDDDDDD    FF          RR      RR  MM       MM      ....
BBBBBBBB    AA      AA  SSSSSSSS    UUUUUUUUUU  DDDDDDDD    FF          RR      RR  MM       MM      ....

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$$UDF_RM (                          ! Basic READ memory
;     2          0002  0                 IDENT = '1-005'              ! File: BASUDFRM.B32
;     3          0003  0                 ) =
;     4          0004  1 BEGIN
;     5          0005  1 !
;     6          0006  1 !****************************************************************************
;     7          0007  1 !*                                                                          *
;     8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
;     9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
;    10          0010  1 !*  ALL RIGHTS RESERVED.                                                     *
;    11          0011  1 !*                                                                          *
;    12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    13          0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    16          0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    17          0017  1 !*  TRANSFERRED.                                                             *
;    18          0018  1 !*                                                                          *
;    19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    20          0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    21          0021  1 !*  CORPORATION.                                                             *
;    22          0022  1 !*                                                                          *
;    23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
;    25          0025  1 !*                                                                          *
;    26          0026  1 !*                                                                          *
;    27          0027  1 !****************************************************************************
;    28          0028  1 !
;    29          0029  1
;    30          0030  1 !++
;    31          0031  1 ! FACILITY:
;    32          0032  1 !
;    33          0033  1 !       BASIC support library - not user callable
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !
;    37          0037  1 !       Implement UDF level of abstraction - element transmitter for READ.
;    38          0038  1 !
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:
;    41          0041  1 !
;    42          0042  1 !       User access mode - AST reentrant
;    43          0043  1 !
;    44          0044  1 ! AUTHOR: Donald G. Petersen, CREATION DATE: 19-Dec-78
;    45          0045  1 !
;    46          0046  1 ! MODIFIED BY:
;    47          0047  1 !
;    48          0048  1 !       DGP 19-Dec-78, : VERSION 1-001
;    49          0049  1 ! 1-001 - original.  DGP 19-Dec-78
;    50          0050  1 ! 1-002 - Make all routines global.  DGP 20-Dec-78
;    51          0051  1 ! 1-003 - Put code in proper PSECT.  JBS 21-DEC-78
;    52          0052  1 ! 1-004 - Change prefix for stack frame names to BSF$.  JBS 08-FEB-1979
;    53          0053  1 ! 1-005 - Change ISB$L_MAJ_F_PTR to ISB$A_MAJ_F_PTR.  JBS 24-JUL-1979
;    54          0054  1 !--
;    55          0055  1
;    56          0056  1 !<BLF/PAGE>
```

```
;    58          0057  1 !
;    59          0058  1 ! SWITCHES
;    60          0059  1 !
;    61          0060  1
;    62          0061  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    63          0062  1
;    64          0063  1 !
;    65          0064  1 ! LINKAGES
;    66          0065  1 !
;    67          0066  1
;    68          0067  1 REQUIRE 'RTLIN:OTSLNK';                         ! define all linkages
;    69          0496  1
;    70          0497  1 !
;    71          0498  1 ! TABLE OF CONTENTS:
;    72          0499  1 !
;    73          0500  1
;    74          0501  1 FORWARD ROUTINE
;    75          0502  1     BAS$$UDF_RMF0 : JSB_UDF0 NOVALUE,           ! Initialize READ memory
;    76          0503  1     BAS$$UDF_RMF1 : CALL_CCB NOVALUE,           ! element transmitter
;    77          0504  1     BAS$$UDF_RMF9 : JSB_UDF9 NOVALUE;           ! end of memory READ
;    78          0505  1
;    79          0506  1 !
;    80          0507  1 ! INCLUDE FILES:
;    81          0508  1 !
;    82          0509  1
;    83          0510  1 REQUIRE 'RTLML:OTSLUB';                         ! Logical unit block offsets
;    84          0650  1
;    85          0651  1 REQUIRE 'RTLML:OTSISB';                         ! I/O statement block offsets
;    86          0819  1
;    87          0820  1 REQUIRE 'RTLIN:BASFRAME';                       ! Basic frame offsets from R11
;    88          1023  1
;    89          1024  1 REQUIRE 'RTLIN:RTLPSECT';                       ! Declare psects macros
;    90          1119  1
;    91          1120  1 LIBRARY 'RTLSTARLE';
;    92          1121  1
;    93          1122  1 !
;    94          1123  1 ! MACROS:
;    95          1124  1 !
;    96          1125  1 !       NONE
;    97          1126  1 !
;    98          1127  1 ! EQUATED SYMBOLS:
;    99          1128  1 !
;   100          1129  1 !       NONE
;   101          1130  1 !
;   102          1131  1 ! PSECTS:
;   103          1132  1 !
;   104          1133  1 DECLARE_PSECTS (BAS);                           ! Declare PSECTs for BAS facility
;   105          1134  1 !
;   106          1135  1 ! OWN STORAGE:
;   107          1136  1 !
;   108          1137  1 !       NONE
;   109          1138  1 !
;   110          1139  1 ! EXTERNAL REFERENCES:
;   111          1140  1 !
;   112          1141  1
;   113          1142  1 EXTERNAL ROUTINE
;   114          1143  1     BAS$$UDF_RL1 : CALL_CCB NOVALUE,
```

```
;  115          1144  1     BAS$$REC_RMF0 : JSB_REC0 NOVALUE,
;  116          1145  1     BAS$$REC_RMF1 : JSB_REC1 NOVALUE,
;  117          1146  1     BAS$$REC_RMF9 : JSB_REC9 NOVALUE;
;  118          1147  1
```

```
;  120          1148  1 GLOBAL ROUTINE BAS$$UDF_RMF0 (FORMAT_ADR) : JSB_UDF0 NOVALUE =  !
;  121          1149  1
;  122          1150  1 !++
;  123          1151  1 ! FUNCTIONAL DESCRIPTION:
;  124          1152  1 !
;  125          1153  1 !       Call REC level of initialization to get all of the buffer pointers set
;  126          1154  1 !       up.
;  127          1155  1 !
;  128          1156  1 !
;  129          1157  1 ! FORMAL PARAMETERS:
;  130          1158  1 !
;  131          1159  1 !       FORMAT_ADR.rl.r         not used
;  132          1160  1 !
;  133          1161  1 ! IMPLICIT INPUTS:
;  134          1162  1 !
;  135          1163  1 !       NONE
;  136          1164  1 !
;  137          1165  1 ! IMPLICIT OUTPUTS:
;  138          1166  1 !
;  139          1167  1 !       NONE
;  140          1168  1 !
;  141          1169  1 ! ROUTINE VALUE:
;  142          1170  1 !
;  143          1171  1 !       NONE
;  144          1172  1 !
;  145          1173  1 ! SIDE EFFECTS:
;  146          1174  1 !
;  147          1175  1 !       NONE
;  148          1176  1 !
;  149          1177  1 !--
;  150          1178  1
;  151          1179  2     BEGIN
;  152          1180  2
;  153          1181  2     EXTERNAL REGISTER
;  154          1182  2         CCB = K_CCB_REG : REF BLOCK [0, BYTE];
;  155          1183  2
;  156          1184  2     !+
;  157          1185  2     ! Call record level routine to initialize the various buffer pointers in
;  158          1186  2     ! the ISB.
;  159          1187  2     !-
;  160          1188  2
;  161          1189  2     BAS$$REC_RMF0 ();
;  162          1190  2     RETURN;
;  163          1191  1     END;                                        !End of BAS$$UDF_RM0


                                                    .TITLE  BAS$$UDF_RM
                                                    .IDENT  \1-005\

                                                    .EXTRN  BAS$$UDF_RL1, BAS$$REC_RMF0
                                                    .EXTRN  BAS$$REC_RMF1, BAS$$REC_RMF9

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

                                00000000G  00  17 00000 BAS$$UDF_RMF0::
                                                    JMP     BAS$$REC_RMF0                                   ; 1189
```

; Routine Size: 6 bytes,   Routine Base: _BAS$CODE + 0000

; 164   1192 1

```
;  166          1193  1 GLOBAL ROUTINE BAS$$UDF_RMF1 (ELEM_TYPE, ELEM_SIZE, ELEM_ADR, FORMAT) : CALL_CCB NOVALUE =      !
;  167          1194  1
;  168          1195  1 !++
;  169          1196  1 ! FUNCTIONAL DESCRIPTION:
;  170          1197  1 !
;  171          1198  1 !       Call BAS$$UDF_RL1.  Then update CUR_DATA in the last major frame upon
;  172          1199  1 !       return.  This done on each element transmitter rather than at the end
;  173          1200  1 !       because there could be an embedded function in the element list which
;  174          1201  1 !       does a READ.
;  175          1202  1 !
;  176          1203  1 !
;  177          1204  1 ! FORMAL PARAMETERS:
;  178          1205  1 !
;  179          1206  1 !       ELEM_TYPE.rlu.v         Type code of user I/O list element
;  180          1207  1 !       ELEM_SIZE.rlu.v         Size of list element
;  181          1208  1 !       ELEM_ADR.rlu.r          Adr of where to store the element
;  182          1209  1 !                               Points to a descriptor for a string
;  183          1210  1 !       FORMAT.rlu.v            Format character following a Prompt string
;  184          1211  1 !
;  185          1212  1 ! IMPLICIT INPUTS:
;  186          1213  1 !
;  187          1214  1 !       ISB$A_MAJ_F_PTR         pointer to last major frame
;  188          1215  1 !       LUB$A_BUF_PTR           pointer in input buffer
;  189          1216  1 !
;  190          1217  1 ! IMPLICIT OUTPUTS:
;  191          1218  1 !
;  192          1219  1 !       BSF$A_CUR_DTA           pointer to current location in DATA area in last
;  193          1220  1 !                               major frame
;  194          1221  1 !
;  195          1222  1 ! ROUTINE VALUE:
;  196          1223  1 !
;  197          1224  1 !       NONE
;  198          1225  1 !
;  199          1226  1 ! SIDE EFFECTS:
;  200          1227  1 !
;  201          1228  1 !       NONE
;  202          1229  1 !
;  203          1230  1 !--
;  204          1231  1
;  205          1232  2     BEGIN
;  206          1233  2
;  207          1234  2     EXTERNAL REGISTER
;  208          1235  2         CCB = K_CCB_REG : REF BLOCK [0, BYTE];
;  209          1236  2
;  210          1237  2     LOCAL
;  211          1238  2         BMF : REF BLOCK [0, BYTE] FIELD (BSF$MAJOR_FRAME);
;  212          1239  2
;  213          1240  2     !+
;  214          1241  2     ! Pick up R11 which points into the last major frame from ISB.  Call
;  215          1242  2     ! BAS$$UDF_RL1 to do the actual element transmit.  Update CUR_DATA in the
;  216          1243  2     ! frame.  Return.
;  217          1244  2     !-
;  218          1245  2
;  219          1246  2     BMF = .CCB [ISB$A_MAJ_F_PTR];
;  220          1247  2     BAS$$UDF_RL1 (.ELEM_TYPE, .ELEM_SIZE, .ELEM_ADR, .FORMAT);
;  221          1248  2
;  222          1249  2     !+
```

```
;  223          1250  2     ! Add one to the current buffer pointer because of the general algorithm
;  224          1251  2     ! INPUT uses.
;  225          1252  2     !-
;  226          1253  2
;  227          1254  2     BMF [BSF$A_CUR_DTA] = .CCB [LUB$A_BUF_PTR] + 1;
;  228          1255  2     RETURN;
;  229          1256  1     END;                                            !End of BAS$$UDF_RMF1


                                        0004 00000      .ENTRY  BAS$$UDF_RMF1, Save R2                    ; 1193
                   52      FF48    CB     D0 00002      MOVL    -184(CCB), BMF                            ; 1246
                   7E        0C    AC     7D 00007      MOVQ    ELEM_ADR, -(SP)                           ; 1247
                   7E        04    AC     7D 0000B      MOVQ    ELEM_TYPE, -(SP)                          ;
            00000000G        00           04 FB 0000F   CALLS   #4, BAS$$UDF_RL1                          ;
   0087   C2       B0        AB           01 C1 00016   ADDL3   #1, -80(CCB), 135(BMF)                    ; 1254
                                             04 0001D   RET                                               ; 1256

; Routine Size:  30 bytes,    Routine Base:  _BAS$CODE + 0006

;  230          1257  1
```

```
;  232          1258  1 GLOBAL ROUTINE BAS$$UDF_RMF9 : JSB_UDF9 NOVALUE =       !
;  233          1259  1
;  234          1260  1 !++
;  235          1261  1 ! FUNCTIONAL DESCRIPTION:
;  236          1262  1 !
;  237          1263  1 !       Call REC level to waste time.
;  238          1264  1 !
;  239          1265  1 !
;  240          1266  1 ! FORMAL PARAMETERS:
;  241          1267  1 !
;  242          1268  1 !       NONE
;  243          1269  1 !
;  244          1270  1 ! IMPLICIT INPUTS:
;  245          1271  1 !
;  246          1272  1 !       NONE
;  247          1273  1 !
;  248          1274  1 ! IMPLICIT OUTPUTS:
;  249          1275  1 !
;  250          1276  1 !       NONE
;  251          1277  1 !
;  252          1278  1 ! ROUTINE VALUE:
;  253          1279  1 !
;  254          1280  1 !       NONE
;  255          1281  1 !
;  256          1282  1 ! SIDE EFFECTS:
;  257          1283  1 !
;  258          1284  1 !       NONE
;  259          1285  1 !
;  260          1286  1 !--
;  261          1287  1
;  262          1288  2     BEGIN
;  263          1289  2
;  264          1290  2     EXTERNAL REGISTER
;  265          1291  2         CCB = K_CCB_REG : REF BLOCK [0, BYTE];
;  266          1292  2
;  267          1293  2     !+
;  268          1294  2     ! Call record level
;  269          1295  2     !-
;  270          1296  2
;  271          1297  2     BAS$$REC_RMF9 ();
;  272          1298  2     RETURN;
;  273          1299  1     END;                                    !End of BAS$$UDF_RMF9


                                00000000G  00  17 00000 BAS$$UDF_RMF9::
                                                        JMP     BAS$$REC_RMF9                   ; 1297

; Routine Size:  6 bytes,    Routine Base:  _BAS$CODE + 0024

;  274          1300  1
;  275          1301  1 END                                     !End of module BAS$$UDF_RM
;  276          1302  1
;  277          1303  0 ELUDOM
```

```
;                                   PSECT SUMMARY
;
;       Name                     Bytes                         Attributes
;
;  _BAS$CODE                           42  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         0         0        581         00:01.2



;                                   COMMAND QUALIFIERS

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

; Size:          42 code + 0 data bytes
; Run Time:         00:09.6
; Elapsed Time:     00:21.7
; Lines/CPU Min:     8135
; Lexemes/CPU-Min: 42199
; Memory Used:  115 pages
; Compilation Complete
```

BASUDFWF
LIS

BASSTRING
LIS

BASTERMIO
LIS

BASTRM
LIS

BASTAB
LIS

BASUDFRM
LIS

BASSYS
LIS

BASSTR
LIS

BASUDFRL
LIS