M 3

```
PPPPPPPPPPPP     RRRRRRRRRRRR     TTTTTTTTTTTTTTT     SSSSSSSSSSSS  MMM         MMM  BBBBBBBBBBBB
PPPPPPPPPPPP     RRRRRRRRRRRR     TTTTTTTTTTTTTTT     SSSSSSSSSSSS  MMM         MMM  BBBBBBBBBBBB
PPPPPPPPPPPP     RRRRRRRRRRRR     TTTTTTTTTTTTTTT     SSSSSSSSSSSS  MMM         MMM  BBBBBBBBBBBB
PPP         PPP  RRR         RRR        TTT        SSS              MMMMMM   MMMMMM  BBB         BBB
PPP         PPP  RRR         RRR        TTT        SSS              MMMMMM   MMMMMM  BBB         BBB
PPP         PPP  RRR         RRR        TTT        SSS              MMMMMM   MMMMMM  BBB         BBB
PPP         PPP  RRR         RRR        TTT        SSS              MMM   MMM   MMM  BBB         BBB
PPP         PPP  RRR         RRR        TTT        SSS              MMM   MMM   MMM  BBB         BBB
PPP         PPP  RRR         RRR        TTT        SSS              MMM   MMM   MMM  BBB         BBB
PPPPPPPPPPPP     RRRRRRRRRRRR           TTT           SSSSSSSSS     MMM         MMM  BBBBBBBBBBBB
PPPPPPPPPPPP     RRRRRRRRRRRR           TTT           SSSSSSSSS     MMM         MMM  BBBBBBBBBBBB
PPPPPPPPPPPP     RRRRRRRRRRRR           TTT           SSSSSSSSS     MMM         MMM  BBBBBBBBBBBB
PPP              RRR   RRR              TTT                    SSS  MMM         MMM  BBB         BBB
PPP              RRR   RRR              TTT                    SSS  MMM         MMM  BBB         BBB
PPP              RRR   RRR              TTT                    SSS  MMM         MMM  BBB         BBB
PPP              RRR      RRR           TTT                    SSS  MMM         MMM  BBB         BBB
PPP              RRR      RRR           TTT                    SSS  MMM         MMM  BBB         BBB
PPP              RRR      RRR           TTT                    SSS  MMM         MMM  BBB         BBB
PPP              RRR         RRR        TTT        SSSSSSSSSSSS     MMM         MMM  BBBBBBBBBBBB
PPP              RRR         RRR        TTT        SSSSSSSSSSSS     MMM         MMM  BBBBBBBBBBBB
PPP              RRR         RRR        TTT        SSSSSSSSSSSS     MMM         MMM  BBBBBBBBBBBB
```

```
**FILE**ID**SEPARATE                                   D 13



  SSSSSSSS  EEEEEEEEEE  PPPPPPPP      AAAAAA    RRRRRRRR      AAAAAA    TTTTTTTTTT  EEEEEEEEEE
  SSSSSSSS  EEEEEEEEEE  PPPPPPPP      AAAAAA    RRRRRRRR      AAAAAA    TTTTTTTTTT  EEEEEEEEEE
SS          EE          PP      PP  AA      AA  RR      RR  AA      AA      TT      EE
SS          EE          PP      PP  AA      AA  RR      RR  AA      AA      TT      EE
SS          EE          PP      PP  AA      AA  RR      RR  AA      AA      TT      EE
SS          EE          PP      PP  AA      AA  RR      RR  AA      AA      TT      EE
  SSSSSS    EEEEEEEE    PPPPPPPP    AA      AA  RRRRRRRR    AA      AA      TT      EEEEEEEE
  SSSSSS    EEEEEEEE    PPPPPPPP    AA      AA  RRRRRRRR    AA      AA      TT      EEEEEEEE
        SS  EE          PP          AAAAAAAAAA  RR  RR      AAAAAAAAAA      TT      EE
        SS  EE          PP          AAAAAAAAAA  RR  RR      AAAAAAAAAA      TT      EE
        SS  EE          PP          AA      AA  RR    RR    AA      AA      TT      EE              ....
        SS  EE          PP          AA      AA  RR    RR    AA      AA      TT      EE              ....
SSSSSSSS    EEEEEEEEEE  PP          AA      AA  RR      RR  AA      AA      TT      EEEEEEEEEE      ....
SSSSSSSS    EEEEEEEEEE  PP          AA      AA  RR      RR  AA      AA      TT      EEEEEEEEEE      ....

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

SEP
V04

; R

```
0 MODULE SEPARATE ( %TITLE 'Print Symbiont -- separation routines'
0         IDENT = 'V04-001',
0         ADDRESSING_MODE (EXTERNAL = GENERAL)
0         ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:
1 !       Print Symbiont.
1 !
1 ! ABSTRACT:
1 !       File and module input routines.
1 !
1 ! ENVIRONMENT:
1 !       VAX/VMS user mode.
1 !--
1 !
1 ! AUTHOR: Rowland R. Bradley
1 !
1 ! CREATION DATE: April 1, 1984
1 !
1 ! MODIFIED BY:
1 !
1 !       40-001  RRB0006         Rowland R. Bradley      14-Aug-1984
1 !               Add a <CR> to  the page header.  This fixes QAR 0682
1 !               QAR 1737, loss  of first line of data of every page.
1 !
1 !       3B-005  RRB0005         Rowland R. Bradley      02-Aug-1984
1 !               Remove the phrase "in this file" in (file) record description.
1 !               Remove extra spaces in job description after start date
1 !               and after queued to date.  Slightly alter filename algorithm to
1 !               avoid placing long filename on a single line and NOT displaying
```

```
;   58      0058  1 !                   the type and version (or vice versa).  Add a call to
;   59      0059  1 !                   PSM$READ_ITEM_DX in GET_QUALIFIERS to test correctness(this
;   60      0060  1 !                   is my call and will remain as a test).  Modify the file trailer
;   61      0061  1 !                   page to avoid an ugly truncation of the input file.  Add the
;   62      0062  1 !                   acronym "UIC" to the job description sentence.  Add /FEED and
;   63      0063  1 !                   /NOFEED to the list of qualifiers in GET_QUALIFIERS.
;   64      0064  1 !
;   65      0065  1 !
;   66      0066  1 !           3B-004  RRB0004         Rowland R. Bradley      04-May-1984
;   67      0067  1 !                   Avoid truncation of the Job Flag Page when burst pages
;   68      0068  1 !                   are not specified.
;   69      0069  1 !
;   70      0070  1 !           3B-003  GRR0003         Gregory R. Robert       29-Apr-1984
;   71      0071  1 !                   Removed reference to JBC$_JOBREQUEUE until job controller
;   72      0072  1 !                   message definitions appear in LIB.L32.  Changed ABORTED
;   73      0073  1 !                   and REQUEUED words in job sentence to uppercase for emphasis.
;   74      0074  1 !                   Considered left and right margins in computing page header
;   75      0075  1 !                   size.  Create page header only once per task.  Fix page header
;   76      0076  1 !                   to have file revision date instead of revision number.
;   77      0077  1 !
;   78      0078  1 !           3B-002  RRB0002         Rowland R. Bradley      27-Apr-1984
;   79      0079  1 !                   Add dynamic Page_Header routine, display all queue qualifiers
;   80      0080  1 !                   in a single phrase, fix form feed and line one problems, print
;   81      0081  1 !                   /setup_file & /setup_form,  insert job number in burst chars,
;   82      0082  1 !                   FIX: footer bar, devicename, receipt box, file desc sentence,
;   83      0083  1 !                        trailer header bar, digital_logo dynamics, /width,
;   84      0084  1 !                        psm$announce display size, tab valid bit, page setup quals.
;   85      0085  1 !
;   86      0086  1 !           3B-001  RRB0001         Rowland R. Bradley      01-Apr-1984
;   87      0087  1 !                   Original version
;   88      0088  1 !
;   89      0089  1 !**
;   91          0090   1 LIBRARY 'SYS$LIBRARY:LIB';
;   92          0091   1 REQUIRE 'LIB$:SMBDEF';
;   93          0583   1 REQUIRE 'SRC$:SMBREQ';
;   94          1040   1
;   95          1041   1 EXTERNAL ROUTINE
;   96          1042   1          PSM$BANNER,
;   97          1043   1          PSM$READ_ITEM_DX                           ! strictly to test behavior
;   98          1044   1          ;
;   99          1045   1
;  100          1046   1 LITERAL
;  101          1047   1          UPCASE_MASK= %B '0100000',                 ! lowercase to uppercase
;  102          1048   1          LEAD_MASK  = %B '001000',                  ! delete leading blanks
;  103          1049   1          SIZE       = 0,
;  104          1050   1          ADDR       = 1
;  105          1051   1          ;
;  106          1052   1
;  107          1053   1 OWN
;  108          1054   1          BURST
;  109          1055   1          ;
;  111      1056  1 FORWARD ROUTINE
;  112      1057  1         PSM$FILE_BURST           ,
;  113      1058  1         PSM$FILE_FLAG            ,
;  114      1059  1         PSM$FILE_TRAILER         ,
;  115      1060  1         PSM$JOB_BURST            ,
;  116      1061  1         PSM$JOB_FLAG             ,
;  117      1062  1         PSM$JOB_TRAILER          ,
;  118      1063  1         PSM$PAGE_HEADER          ,
;  119      1064  1
;  120      1065  1         PARSE_FILE_NAME          ,
;  121      1066  1         ALLOCATE_PAGE            ,
;  122      1067  1         DEALLOCATE_PAGE          ,
;  123      1068  1         CREATE_PAGE_HEADER       ,
;  124      1069  1         RETURN_FRAME_LENGTH      ,
;  125      1070  1         RETURN_FRAME_WIDTH       ,
;  126      1071  1         FILL_JOB_FLAG            : NOVALUE,
;  127      1072  1         FILL_FILE_FLAG           : NOVALUE,
;  128      1073  1         FILL_JOB_TRAILER         : NOVALUE,
;  129      1074  1         FILL_FILE_TRAILER        : NOVALUE,
;  130      1075  1         GET_FORM_SIZE            : NOVALUE,
;  131      1076  1         GET_REVISION_DATE        : NOVALUE,
;  132      1077  1         GET_SYSTEM_ANNOUNCEMENT  : NOVALUE,
;  133      1078  1         GET_JOB_DESCRIPTION      : NOVALUE,
;  134      1079  1         GET_FILE_DESCRIPTION     : NOVALUE,
;  135      1080  1         GET_FILE_NAME            : NOVALUE,
;  136      1081  1         GET_JOB_NAME             : NOVALUE,
;  137      1082  1         GET_EOJ                  : NOVALUE,
;  138      1083  1         GET_EOF                  : NOVALUE,
;  139      1084  1         GET_ACCOUNTING_INFO      : NOVALUE,
;  140      1085  1         GET_QUALIFIERS           : NOVALUE,
;  141      1086  1         GET_QUEUE_QUALIFIERS     : NOVALUE,
;  142      1087  1         GET_USER_NOTE            : NOVALUE,
;  143      1088  1         GET_RECEIPT_BOX          : NOVALUE,
;  144      1089  1         GET_VMS_LOGO             : NOVALUE,
;  145      1090  1         GET_DIGITAL_LOGO         : NOVALUE,
;  146      1091  1         GET_RULER_FINE           : NOVALUE,
;  147      1092  1         GET_RULER_COARSE         : NOVALUE,
;  148      1093  1         SCROLL_FRAME             : NOVALUE,      ! scrolls/fills frame w/ string
;  149      1094  1         FILL_FRAME               : NOVALUE,      ! fills a frame with chars
;  150      1095  1         MOVE_FRAME               : NOVALUE,      ! inserts undelimited strings
;  151      1096  1         INSERT_FRAME             : NOVALUE,      ! inserts delimited strings
;  152      1097  1         CENTER_FRAME             : NOVALUE,      ! centers undelimited strings
;  153      1098  1         MERGE_FRAME              : NOVALUE,      ! merges "pure" frames
;  154      1099  1         INSERT_NAME_BANNER       ,               ! inserts banners into frames
;  155      1100  1         INSERT_FILENAME_BANNER   ,               ! inserts banners into frames
;  156      1101  1         INSERT_JOBNUMBER_BANNER  ,               ! inserts banners into frames
;  157      1102  1         DELIMIT_STRING           ,               ! delimits strings
;  158      1103  1         DELIMIT_STRING_NOT       ,               ! delimits strings
;  159      1104  1         FIND_DEST_PTR            : NOVALUE,      ! identifies "pure" frames
;  160      1105  1         FIND_SOURCE_PTR          : NOVALUE,      ! identifies "impure" frames
;  161      1106  1         DISCARD                  : NOVALUE,      ! discards chars in strings
;  162      1107  1         FILE_OPEN                                ! determines file accessibility
;  163      1108  1         ;
;  164      1109  1
```

```
1 %sbttl 'PSM$FILE_BURST - Print a File Burst Page'
1 ! Functional Description:
1 !         This routine controls the creation of the file burst page.  The
1 !         FUNCTION code dictates the action taken in creation.
1 !         FUNCTION:
1 !                 OPEN  - Allocate and create the File Burst Page
1 !                 READ  - Return the current line of the File Burst Page
1 !                 CLOSE - Return the buffer allocated on OPEN
1 !
1 ! Formal Parameters:
1 !                 SMB_CONTEXT   - Pointer to the SMB
1 !                 USER_CONTEXT  - User defined pointer (not used here)
1 !                 FUNCTION      - OPEN, READ, CLOSE
1 !                 FUNC_DESC     - Pointer to functionally dependent descriptor
1 !                 FUNC_ARG      - Pointer to functionally dependent argument
1 !
1 ! Implicit Inputs:
1 !                 none
1 !
1 ! Implicit Outputs:
1 !                 none
1 !
1 ! Returned Value:
1 !                 none
1 !
1 ! Side Effects:
1 !                 none
1 !--
1 GLOBAL ROUTINE PSM$FILE_BURST ( %SBTTL 'FILE_BURST'
1         SMB_CONTEXT     : REF VECTOR,
1         USER_CONTEXT    : REF VECTOR,
1         FUNCTION        : REF VECTOR,
1         FUNC_DESC       : REF VECTOR,
1         FUNC_ARG        : REF VECTOR
1         )       =
2 BEGIN
2
2 LOCAL
2         SCB             : REF $BBLOCK,
2         STATUS,
2         FORM_WIDTH,
2         FORM_LENGTH,
2         FORM_SIZE,
2         PAGE_REF        : REF PAGE_ARRAY,       ! Declare the pointer to page
2         BUFFER          : VECTOR [512,byte],    ! Assume max size 512 bytes
2         STRING_DESC     : VECTOR [2];           ! Descriptor to current string
2
2
2 SCB = .SMB_CONTEXT[0];
2
2 ! Check the FUNCTION requested
2
2 SELECTONEU .FUNCTION[0] OF
2 SET
2     [PSM$K_READ]:
3         BEGIN
3         PAGE_REF = .SCB[PSM$A_PAGE_POINTER];
```

```
;  223   1167 3                                                    !  Output one line at a time
;  224   1168 3         IF .SCB[PSM$L_RECORD_NUMBER] GTR (.SCB[PSM$L_PAGE_LENGTH]-6) THEN
;  225   1169 3             RETURN PSM$_EOF;
;  226   1170 3
;  227   1171 3         FUNC_DESC[SIZE] = .SCB[PSM$L_PAGE_WIDTH];
;  228   1172 3         FUNC_DESC[ADDR] = PAGE_REF[0,.SCB[PSM$L_RECORD_NUMBER],
;  229   1173 3                                    .SCB[PSM$L_PAGE_WIDTH]];
;  230   1174 3
;  231   1175 3                                  ! adjust pointer
;  232   1176 3         FUNC_DESC[SIZE] = DELIMIT_STRING_NOT (.FUNC_DESC[ADDR],
;  233   1177 3                                              %CHAR(32), .FUNC_DESC[SIZE]);
;  234   1178 3
;  235   1179 2         END;
;  236   1180 2
;  237   1181 2     [PSM$K_OPEN]:
;  238   1182 3         BEGIN
;  239   1183 3
;  240   1184 3         GET_FORM_SIZE (.SCB);                           ! Returns the WidthxLength
;  241   1185 3
;  242   1186 3
;  243   1187 3         FORM_WIDTH      = .SCB[PSM$L_PAGE_WIDTH];
;  244   1188 3         FORM_LENGTH     = .SCB[PSM$L_PAGE_LENGTH];
;  245   1189 3
;  246   1190 3         RETURN_IF_ERROR_(ALLOCATE_PAGE( .SCB)); ! Get the page of memory
;  247   1191 3
;  248   1192 3         PAGE_REF = .SCB[PSM$A_PAGE_POINTER];    ! My local page pointer
;  249   1193 3
;  250   1194 3         ! Allocate the buffer for "GET_xxx" Routines
;  251   1195 3         !
;  252   1196 3         STRING_DESC[SIZE] = %ALLOCATION(BUFFER);        ! allocate for routines
;  253   1197 3         STRING_DESC[ADDR] = BUFFER;                     ! init address
;  254   1198 3
;  255   1199 3         ! No Form_feed for the burst page
;  256   1200 3
;  257   1201 3         ! Format the page identically to File Flag
;  258   1202 3         !  Standard Burst Page 132x66:  text covers rows 2 through 60,
;  259   1203 3         !    translated to frames... ref starts at 2 and length is 58.
;  260   1204 3         FILL_FILE_FLAG(
;  261   1205 3                     .SCB,
;  262   1206 3                     PAGE_REF[0,2,.FORM_WIDTH],
;  263   1207 3                     .FORM_WIDTH,
;  264   1208 3                     .FORM_LENGTH - 6 - 2 );             ! bottom margin is 6
;  265   1209 3                                                         ! 2 spaces at the top
;  266   1210 2         END;
;  267   1211 2     [PSM$K_CLOSE]:                      ! Return the Page of Memory
;  268   1212 2         RETURN_IF_ERROR_(DEALLOCATE_PAGE(.SCB));
;  269   1213 2
;  270   1214 2     [OTHERWISE]:
;  271   1215 2         RETURN PSM$_FUNNOTSUP;
;  272   1216 2
;  273   1217 2     TES; ! case .function
;  274   1218 2
;  275   1219 2 SS$_NORMAL
;  276   1220 2
;  277   1221 1 END;
```

```
                                                         .TITLE  SEPARATE Print Symbiont -- separation routines
                                                         .IDENT  \V04-001\

                                                         .PSECT  DATA,NOEXE,2

                                        00000 BURST:     .BLKB   4

                                                         .EXTRN  BAS$EDIT, LBR$CLOSE
                                                         .EXTRN  LBR$GET_RECORD, LBR$INI_CONTROL
                                                         .EXTRN  LBR$LOOKUP_KEY, LBR$OPEN
                                                         .EXTRN  LBR$RET_RMSSTV, LBR$SET_LOCATE
                                                         .EXTRN  LIB$TRIM_FILESPEC
                                                         .EXTRN  LIB$GET_VM, LIB$FREE_VM
                                                         .EXTRN  STR$ANALYZE_SDESC
                                                         .EXTRN  STR$ANALYZE_SDESC_R1
                                                         .EXTRN  STR$APPEND, STR$CONCAT
                                                         .EXTRN  STR$COPY_DX, STR$COPY_R
                                                         .EXTRN  STR$FREE1_DX, STR$FREE1_DX_R4
                                                         .EXTRN  STR$GET1_DX, STR$LEFT
                                                         .EXTRN  STR$PREFIX, STR$RIGHT
                                                         .EXTRN  PSM$S_HANGUP_DISPATCH_ENTRY
                                                         .EXTRN  PSM$_BUFFEROVF, PSM$_EOF
                                                         .EXTRN  PSM$_ESCAPE, PSM$_FLUSH
                                                         .EXTRN  PSM$_FUNNOTSUP, PSM$_INVITMCOD
                                                         .EXTRN  PSM$_INVVMSOSC, PSM$_MODNOTFND
                                                         .EXTRN  PSM$_NEWPAGE, PSM$_NOFILEID
                                                         .EXTRN  PSM$_OSCTOOLON, PSM$_PENDING
                                                         .EXTRN  PSM$_SUSPEND, PSM$_TOOMANYLEV
                                                         .EXTRN  SMB$_INVSTMNBR, SMB$_INVSTRLEV
                                                         .EXTRN  SMB$_NOMOREITEMS
                                                         .EXTRN  PSM$BANNER, PSM$READ_ITEM_DX

                                                         .PSECT  CODE,NOWRT,2

                                 003C 00000              .ENTRY  PSM$FILE_BURST, Save R2,R3,R4,R5                    ; 1138
              5E      FDF8    CE   9E 00002              MOVAB   -520(SP), SP                                         ;
              52        04    BC   D0 00007              MOVL    @SMB_CONTEXT, SCB                                    ; 1158
              50        0C    BC   D0 0000B              MOVL    @FUNCTION, R0                                        ; 1162
              05              50   D1 0000F              CMPL    R0, #5                                               ; 1164
                              41   12 00012              BNEQ    2$                                                   ;
              55      01FC    C2   D0 00014              MOVL    508(SCB), PAGE_REF                                   ; 1166
     50  01F8 C2              06   C3 00019              SUBL3   #6, 504(SCB), R0                                     ; 1168
              50      026C    C2   D1 0001F              CMPL    620(SCB), R0                                         ;
                              08   15 00024              BLEQ    1$                                                   ;
              50 00000000G    8F   D0 00026              MOVL    #PSM$_EOF, R0                                        ; 1169
                                   04 0002D              RET                                                          ;
              53        10    AC   D0 0002E 1$:          MOVL    FUNC_DESC, R3                                        ; 1171
              63      0200    C2   D0 00032              MOVL    512(SCB), (R3)                                       ;
     50  026C C2      0200    C2   C5 00037              MULL3   512(SCB), 620(SCB), R0                               ; 1173
 04  A3       50              55   C1 0003F              ADDL3   PAGE_REF, R0, 4(R3)                                  ;
                              63   DD 00044              PUSHL   (R3)                                                 ; 1177
                              20   DD 00046              PUSHL   #32                                                  ; 1176
                        04    A3   DD 00048              PUSHL   4(R3)                                                ;
         0000V CF             03   FB 0004B              CALLS   #3, DELIMIT_STRING_NOT                               ;
              63              50   D0 00050              MOVL    R0, (R3)                                             ;
                              58   11 00053              BRB     5$                                                   ; 1162
              04              50   D1 00055 2$:          CMPL    R0, #4                                               ; 1181
```

```
                                             3B 12 00058        BNEQ     3$                                  ;
                                             52 DD 0005A        PUSHL    SCB                                 ; 1184
                    0000V  CF                01 FB 0005C        CALLS    #1, GET_FORM_SIZE                   ;
                           53        0200    C2 D0 00061        MOVL     512(SCB), FORM_WIDTH                ; 1187
                           54        01F8    C2 D0 00066        MOVL     504(SCB), FORM_LENGTH               ; 1188
                                             52 DD 0006B        PUSHL    SCB                                 ; 1190
                    0000V  CF                01 FB 0006D        CALLS    #1, ALLOCATE_PAGE                   ;
                           3B                50 E9 00072        BLBC     STATUS, 6$                          ;
                           55        01FC    C2 D0 00075        MOVL     508(SCB), PAGE_REF                  ; 1192
                           6E        0200    8F 3C 0007A        MOVZWL   #512, STRING_DESC                   ; 1196
                       04  AE          08    AE 9E 0007F        MOVAB    BUFFER, STRING_DESC+4               ; 1197
                                       F8    A4 9F 00084        PUSHAB   -8(FORM_LENGTH)                     ; 1208
                                             53 DD 00087        PUSHL    FORM_WIDTH                          ; 1207
                                           6543 3F 00089        PUSHAW   (PAGE_REF)[FORM_WIDTH]              ; 1206
                                             52 DD 0008C        PUSHL    SCB                                 ;
                    0000V  CF                04 FB 0008E        CALLS    #4, FILL_FILE_FLAG                  ;
                                             18 11 00093        BRB      5$                                  ; 1162
                           02                50 D1 00095 3$:    CMPL     R0, #2                              ; 1211
                                             0B 12 00098        BNEQ     4$                                  ;
                                             52 DD 0009A        PUSHL    SCB                                 ; 1212
                    0000V  CF                01 FB 0009C        CALLS    #1, DEALLOCATE_PAGE                 ;
                           09                50 E8 000A1        BLBS     STATUS, 5$                          ;
                                                04 000A4        RET                                          ;
                           50 00000000G      8F D0 000A5 4$:    MOVL     #PSM$_FUNNOTSUP, R0                 ; 1215
                                                04 000AC        RET                                          ;
                           50                01 D0 000AD 5$:    MOVL     #1, R0                              ; 1221
                                                04 000B0 6$:    RET                                          ;

; Routine Size:  177 bytes,    Routine Base:  CODE + 0000
```

```
1 %sbttl 'PSM$FILE_FLAG - Print a File Flag Page'
1 ! Functional Description:
1 !         This routine controls the creation of the file flag page.  The
1 !         FUNCTION code dictates the action taken in creation.
1 !         FUNCTION:
1 !                 OPEN  - Allocate and create the file Flag Page
1 !                 READ  - Return the current line of the file Flag Page
1 !                 CLOSE - Return the buffer allocated on OPEN
1 !
1 ! Formal Parameters:
1 !                 SMB_CONTEXT   - Pointer to the SMB
1 !                 USER_CONTEXT  - User defined pointer (not used here)
1 !                 FUNCTION      - OPEN, READ, CLOSE
1 !                 FUNC_DESC     - Pointer to functionally dependent descriptor
1 !                 FUNC_ARG      - Pointer to functionally dependent argument
1 !
1 ! Implicit Inputs:
1 !                 none
1 !
1 ! Implicit Outputs:
1 !                 none
1 !
1 ! Returned Value:
1 !                 none
1 !
1 ! Side Effects:
1 !                 none
1 !--
1
1 GLOBAL ROUTINE PSM$FILE_FLAG ( %SBTTL 'FILE_FLAG'
1         SMB_CONTEXT       : REF VECTOR,
1         USER_CONTEXT      : REF VECTOR,
1         FUNCTION          : REF VECTOR,
1         FUNC_DESC         : REF VECTOR,
1         FUNC_ARG          : REF VECTOR
1         )       =
2 BEGIN
2
2 LOCAL
2         SCB               : REF $BBLOCK,
2         STATUS,
2         FORM_WIDTH,
2         FORM_LENGTH,
2         FORM_SIZE,
2         BUFFER            : VECTOR [512,byte],    ! Assume max size 512 bytes
2         STRING_DESC       : VECTOR [2],           ! Descriptor to current string
2         PAGE_REF          : REF PAGE_ARRAY;       ! Declare the pointer
2                                                   !  to page
2
2 SCB = .SMB_CONTEXT[0];
2
2 ! Check the FUNCTION requested
2
2 SELECTONEU .FUNCTION[0] OF
2 SET
2     [PSM$K_READ]:
3         BEGIN
```

```
3               PAGE_REF = .SCB[PSM$A_PAGE_POINTER];
3                                                       !   Output one line at a time
4               IF (   .SCB[PSM$L_RECORD_NUMBER] GTR .SCB[PSM$L_PAGE_LENGTH])
3                               OR
4                 ( NOT .SEPARATE_FLAG_(FILE_BURST) AND
4               .SCB[PSM$L_RECORD_NUMBER] GTR (.SCB[PSM$L_PAGE_LENGTH] - 6) )
3                       THEN
3                   RETURN PSM$_EOF;
3
3               FUNC_DESC[SIZE] = .SCB[PSM$L_PAGE_WIDTH];
3               FUNC_DESC[ADDR] = PAGE_REF[0,.SCB[PSM$L_RECORD_NUMBER],
3                                               .SCB[PSM$L_PAGE_WIDTH]];
3
3                                               ! adjust pointer
3               FUNC_DESC[SIZE] = DELIMIT_STRING_NOT (.FUNC_DESC[ADDR],
3                                                       %CHAR(32), .FUNC_DESC[SIZE]);
3
2               END;
2
2           [PSM$K_OPEN]:
3               BEGIN
3
3               GET_FORM_SIZE (.SCB);                                   ! Returns the WidthxLength
3
3
3               FORM_WIDTH      = .SCB[PSM$L_PAGE_WIDTH];
3               FORM_LENGTH     = .SCB[PSM$L_PAGE_LENGTH];
3
3               RETURN_IF_ERROR_(ALLOCATE_PAGE( .SCB));                 ! Get the page of memory
3
3               PAGE_REF = .SCB[PSM$A_PAGE_POINTER];                    ! My local page pointer
3
3               ! Always start at top of page
3               !
3               PAGE_REF[0,0,.FORM_WIDTH] = PSM$K_CHAR_FF;              ! form feed in 0 pos.
3
3               !  Standard Flag Page 132x66:  text covers rows 1 through 58,
3               !     translated to frames... ref starts at 1 and length is 57.
3               FILL_FILE_FLAG(  .SCB,
3                               PAGE_REF[0,1,.FORM_WIDTH],
3                               .FORM_WIDTH,
3                               .FORM_LENGTH - 6 - 2 - 1);                      ! total form length...
3                                                                       !   6 burst, 2 spaces
3                                                                       !   top margin of 1
3               ! Burst always appears on the Flag page, starting at page length - 5
3               !   and contiuing to page length.  This leaves a two blank lines
3               !   between File flag footer bar and file burst header bar.
3               !   This IS the right way to perform a BURST over the crease !
3               !
3               IF (.SEPARATE_FLAG_(FILE_BURST)) THEN
4                   BEGIN
4                   STRING_DESC[SIZE] = %ALLOCATION(BUFFER);
4                   STRING_DESC[ADDR] = BUFFER;
4
4                   ! determine the correct size of the string to insert
4                   !
4                   GET_VMS_LOGO
```

```
;  393          1336  4                    (.SCB,
;  394          1337  4                    STRING_DESC[0],                  ! Buffer descriptor
;  395          1338  4                    STRING_DESC[SIZE]);              ! Returned length
;  396          1339  4
;  397          1340  4            INSERT_FRAME
;  398          1341  4                    (.SCB,
;  399          1342  4                    STRING_DESC[0],
;  400          1343  4                    PAGE_REF[10,.FORM_LENGTH-5,.FORM_WIDTH],
;  401          1344  4                    .FORM_WIDTH-20, 1];
;  402          1345  4            INSERT_FRAME
;  403          1346  4                    (.SCB,
;  404          1347  4                    STRING_DESC[0],
;  405          1348  4                    PAGE_REF[14,.FORM_LENGTH-4,.FORM_WIDTH],
;  406          1349  4                    .FORM_WIDTH-16, 1];
;  407          1350  4            INSERT_FRAME
;  408          1351  4                    (.SCB,
;  409          1352  4                    STRING_DESC[0],
;  410          1353  4                    PAGE_REF[10,.FORM_LENGTH-3,.FORM_WIDTH],
;  411          1354  4                    .FORM_WIDTH-20, 1];
;  412          1355  4            INSERT_FRAME
;  413          1356  4                    (.SCB,
;  414          1357  4                    STRING_DESC[0],
;  415          1358  4                    PAGE_REF[14,.FORM_LENGTH-2,.FORM_WIDTH],
;  416          1359  4                    .FORM_WIDTH-16, 1];
;  417          1360  4            INSERT_FRAME
;  418          1361  4                    (.SCB,
;  419          1362  4                    STRING_DESC[0],
;  420          1363  4                    PAGE_REF[10,.FORM_LENGTH-1,.FORM_WIDTH],
;  421          1364  4                    .FORM_WIDTH-20, 1];
;  422          1365  4            INSERT_FRAME
;  423          1366  4                    (.SCB,
;  424          1367  4                    STRING_DESC[0],
;  425          1368  4                    PAGE_REF[14,.FORM_LENGTH,.FORM_WIDTH],
;  426          1369  4                    .FORM_WIDTH-16, 1];
;  427          1370  3            END;
;  428          1371  2        END;
;  429          1372  2
;  430          1373  2    [PSM$K_CLOSE]:                      ! Return the Page of Memory
;  431          1374  2        RETURN_IF_ERROR_(DEALLOCATE_PAGE(.SCB));
;  432          1375  2
;  433          1376  2    [OTHERWISE]:
;  434          1377  2        RETURN PSM$_FUNNOTSUP;
;  435          1378  2
;  436          1379  2    TES; ! case .function
;  437          1380  2
;  438          1381  2 SS$_NORMAL
;  439          1382  2
;  440          1383  1 END;



                                       007C 00000          .ENTRY  PSM$FILE_FLAG, Save R2,R3,R4,R5,R6                 ; 1251
                        56     0000V  CF   9E 00002          MOVAB   INSERT_FRAME, R6                                    ;
                        5E      FDF8  CE   9E 00007          MOVAB   -520(SP), SP                                        ;
                        54        04  BC   D0 0000C          MOVL    @SMB_CONTEXT, SCB                                   ; 1271
```

```
                      50          0C    BC D0 00010       MOVL    @FUNCTION, R0                   ; 1275
                      05                50 D1 00014       CMPL    R0, #5                          ; 1277
                                        4E 12 00017       BNEQ    4$                              ;
                      53        01FC    C4 D0 00019       MOVL    508(SCB), PAGE_REF              ; 1279
                      50        026C    C4 D0 0001E       MOVL    620(SCB), R0                    ; 1281
                01F8    C4              50 D1 00023       CMPL    R0, 504(SCB)                    ;
                                        10 14 00028       BGTR    1$                              ;
                      13        0154    C4 E8 0002A       BLBS    340(SCB), 2$                    ; 1283
          51    01F8    C4              06 C3 0002F       SUBL3   #6, 504(SCB), R1                ; 1284
                      51                50 D1 00035       CMPL    R0, R1                          ;
                                        08 15 00038       BLEQ    2$                              ;
                      50   00000000G    8F D0 0003A 1$:   MOVL    #PSM$_EOF, R0                   ; 1286
                                           04 00041       RET                                     ;
                      52          10    AC D0 00042 2$:   MOVL    FUNC_DESC, R2                   ; 1288
                      62        0200    C4 D0 00046       MOVL    512(SCB), (R2)                  ;
                      50        0200    C4 C4 0004B       MULL2   512(SCB), R0                    ; 1290
  04    A2            50                53 C1 00050       ADDL3   PAGE_REF, R0, 4(R2)             ;
                                        62 DD 00055       PUSHL   (R2)                            ; 1294
                                        20 DD 00057       PUSHL   #32                             ; 1293
                                  04    A2 DD 00059       PUSHL   4(R2)                           ;
               0000V    CF              03 FB 0005C       CALLS   #3, DELIMIT_STRING_NOT          ;
                      62                50 D0 00061       MOVL    R0, (R2)                        ;
                                      00FC 31 00064 3$:   BRW     9$                              ; 1275
                      04                50 D1 00067 4$:   CMPL    R0, #4                          ; 1298
                                        03 13 0006A       BEQL    5$                              ;
                                      00DC 31 0006C       BRW     7$                              ;
                                        54 DD 0006F 5$:   PUSHL   SCB                             ; 1301
               0000V    CF              01 FB 00071       CALLS   #1, GET_FORM_SIZE               ;
                      55        0200    C4 D0 00076       MOVL    512(SCB), FORM_WIDTH            ; 1304
                      52        01F8    C4 D0 0007B       MOVL    504(SCB), FORM_LENGTH           ; 1305
                                        54 DD 00080       PUSHL   SCB                             ; 1307
               0000V    CF              01 FB 00082       CALLS   #1, ALLOCATE_PAGE               ;
                      01                50 E8 00087       BLBS    STATUS, 6$                      ;
                                           04 0008A       RET                                     ;
                      53        01FC    C4 D0 0008B 6$:   MOVL    508(SCB), PAGE_REF              ; 1309
                      63                0C 90 00090       MOVB    #12, (PAGE_REF)                 ; 1313
                                  F7    A2 9F 00093       PUSHAB  -9(FORM_LENGTH)                 ; 1320
                                        55 DD 00096       PUSHL   FORM_WIDTH                      ; 1319
                                      6543 9F 00098       PUSHAB  (FORM_WIDTH)[PAGE_REF]          ; 1318
                                        54 DD 0009B       PUSHL   SCB                             ;
               0000V    CF              04 FB 0009D       CALLS   #4, FILL_FILE_FLAG              ;
                      BD        0154    C4 E9 000A2       BLBC    340(SCB), 3$                    ; 1328
                      6E        0200    8F 3C 000A7       MOVZWL  #512, STRING_DESC               ; 1330
                  04    AE          08    AE 9E 000AC       MOVAB   BUFFER, STRING_DESC+4           ; 1331
                                        5E DD 000B1       PUSHL   SP                              ; 1338
                                  04    AE 9F 000B3       PUSHAB  STRING_DESC                     ; 1337
                                        54 DD 000B6       PUSHL   SCB                             ; 1336
               0000V    CF              03 FB 000B8       CALLS   #3, GET_VMS_LOGO                ;
                                        01 DD 000BD       PUSHL   #1                              ; 1343
                                  EC    A5 9F 000BF       PUSHAB  -20(FORM_WIDTH)                 ; 1344
                      50          FB    A2 9E 000C2       MOVAB   -5(R2), R0                      ; 1343
                      50                55 C4 000C6       MULL2   FORM_WIDTH, R0                  ;
                                  0A  A043 9F 000C9       PUSHAB  10(R0)[PAGE_REF]                ;
                                  0C    AE 9F 000CD       PUSHAB  STRING_DESC                     ; 1342
                                        54 DD 000D0       PUSHL   SCB                             ; 1343
                      66                05 FB 000D2       CALLS   #5, INSERT_FRAME                ;
                                        01 DD 000D5       PUSHL   #1                              ; 1348
```

```
                         F0   A5  9F 000D7         PUSHAB  -16(FORM_WIDTH)                  ; 1349
           50            FC   A2  9E 000DA         MOVAB   -4(R2), R0                       ; 1348
           50                 55  C4 000DE         MULL2   FORM_WIDTH, R0                   ;
                         0E A043  9F 000E1         PUSHAB  14(R0)[PAGE_REF]                 ;
                         0C   AE  9F 000E5         PUSHAB  STRING_DESC                      ; 1347
                              54  DD 000E8         PUSHL   SCB                              ; 1348
           66                 05  FB 000EA         CALLS   #5, INSERT_FRAME                 ;
                              01  DD 000ED         PUSHL   #1                               ; 1353
                         EC   A5  9F 000EF         PUSHAB  -20(FORM_WIDTH)                  ; 1354
           50            FD   A2  9E 000F2         MOVAB   -3(R2), R0                       ; 1353
           50                 55  C4 000F6         MULL2   FORM_WIDTH, R0                   ;
                         0A A043  9F 000F9         PUSHAB  10(R0)[PAGE_REF]                 ;
                         0C   AE  9F 000FD         PUSHAB  STRING_DESC                      ; 1352
                              54  DD 00100         PUSHL   SCB                              ; 1353
           66                 05  FB 00102         CALLS   #5, INSERT_FRAME                 ;
                              01  DD 00105         PUSHL   #1                               ; 1358
                         F0   A5  9F 00107         PUSHAB  -16(FORM_WIDTH)                  ; 1359
           50            FE   A2  9E 0010A         MOVAB   -2(R2), R0                       ; 1358
           50                 55  C4 0010E         MULL2   FORM_WIDTH, R0                   ;
                         0E A043  9F 00111         PUSHAB  14(R0)[PAGE_REF]                 ;
                         0C   AE  9F 00115         PUSHAB  STRING_DESC                      ; 1357
                              54  DD 00118         PUSHL   SCB                              ; 1358
           66                 05  FB 0011A         CALLS   #5, INSERT_FRAME                 ;
                              01  DD 0011D         PUSHL   #1                               ; 1363
                         EC   A5  9F 0011F         PUSHAB  -20(FORM_WIDTH)                  ; 1364
           50            FF   A2  9E 00122         MOVAB   -1(R2), R0                       ; 1363
           50                 55  C4 00126         MULL2   FORM_WIDTH, R0                   ;
                         0A A043  9F 00129         PUSHAB  10(R0)[PAGE_REF]                 ;
                         0C   AE  9F 0012D         PUSHAB  STRING_DESC                      ; 1362
                              54  DD 00130         PUSHL   SCB                              ; 1363
           66                 05  FB 00132         CALLS   #5, INSERT_FRAME                 ;
                              01  DD 00135         PUSHL   #1                               ; 1368
                         F0   A5  9F 00137         PUSHAB  -16(FORM_WIDTH)                  ; 1369
           52                 55  C4 0013A         MULL2   FORM_WIDTH, R2                   ; 1368
                         0E A243  9F 0013D         PUSHAB  14(R2)[PAGE_REF]                 ;
                         0C   AE  9F 00141         PUSHAB  STRING_DESC                      ; 1367
                              54  DD 00144         PUSHL   SCB                              ; 1368
           66                 05  FB 00146         CALLS   #5, INSERT_FRAME                 ;
                              18  11 00149         BRB     9$                               ; 1275
           02                 50  D1 0014B 7$:     CMPL    R0, #2                           ; 1373
                              0B  12 0014E         BNEQ    8$                               ;
                              54  DD 00150         PUSHL   SCB                              ; 1374
      0000V  CF               01  FB 00152         CALLS   #1, DEALLOCATE_PAGE              ;
             09               50  E8 00157         BLBS    STATUS, 9$                       ;
                                  04 0015A         RET                                      ;
           50 00000000G  8F       D0 0015B 8$:     MOVL    #PSM$_FUNNOTSUP, R0              ; 1377
                                  04 00162         RET                                      ;
           50                 01  D0 00163 9$:     MOVL    #1, R0                           ; 1383
                                  04 00166         RET                                      ;

; Routine Size:  359 bytes,    Routine Base:  CODE + 00B1
```

```
1 %sbttl 'PSM$FILE_TRAILER - Print a File Trailer Page'
1 ! Functional Description:
1 !       This routine controls the creation of the file flag page.  The
1 !       FUNCTION code dictates the action taken in creation.
1 !       FUNCTION:
1 !               OPEN  - Allocate and create the file Flag Page
1 !               READ  - Return the current line of the file Flag Page
1 !               CLOSE - Return the buffer allocated on OPEN
1 !
1 ! Formal Parameters:
1 !               SMB_CONTEXT   - Pointer to the SMB
1 !               USER_CONTEXT  - User defined pointer (not used here)
1 !               FUNCTION      - OPEN, READ, CLOSE
1 !               FUNC_DESC     - Pointer to functionally dependent descriptor
1 !               FUNC_ARG      - Pointer to functionally dependent argument
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1
1 GLOBAL ROUTINE PSM$FILE_TRAILER ( %SBTTL 'FILE_TRAILER'
1       SMB_CONTEXT     : REF VECTOR,
1       USER_CONTEXT    : REF VECTOR,
1       FUNCTION        : REF VECTOR,
1       FUNC_DESC       : REF VECTOR,
1       FUNC_ARG        : REF VECTOR
1       ) =
2 BEGIN
2
2 LITERAL
2       TRAILING = 1;
2 LOCAL
2       SCB             : REF $BBLOCK,
2       STATUS,
2       FORM_WIDTH,
2       FORM_LENGTH,
2       FORM_SIZE,
2       PAGE_REF        : REF PAGE_ARRAY;       ! Declare the pointer
2                                               !  to page
2
2 SCB = .SMB_CONTEXT[0];
2
2 ! Check the FUNCTION requested
2
2 SELECTONEU .FUNCTION[0] OF
2 SET
2     [PSM$K_READ]:
3       BEGIN
```

```
;  499        1441  3           LOCAL  TEMP_PTR;
;  500        1442  3
;  501        1443  3           PAGE_REF = .SCB[PSM$A_PAGE_POINTER];
;  502        1444  3                                                   !   Output one line at a time
;  503        1445  3           IF .SCB[PSM$L_RECORD_NUMBER] GTR (.SCB[PSM$L_PAGE_LENGTH] - 6 - 2) THEN
;  504        1446  3               RETURN PSM$_EOF;
;  505        1447  3
;  506        1448  3           FUNC_DESC[SIZE] = .SCB[PSM$L_PAGE_WIDTH];
;  507        1449  3           FUNC_DESC[ADDR] = PAGE_REF[0,.SCB[PSM$L_RECORD_NUMBER],
;  508        1450  3                                       .SCB[PSM$L_PAGE_WIDTH]];
;  509        1451  3
;  510        1452  3                                   ! adjust pointer
;  511        1453  3           FUNC_DESC[SIZE] = DELIMIT_STRING_NOT (.FUNC_DESC[ADDR],
;  512        1454  3                                                 %CHAR(32), .FUNC_DESC[SIZE]);
;  513        1455  3
;  514        1456  2           END;
;  515        1457  2
;  516        1458  2       [PSM$K_OPEN]:
;  517        1459  3           BEGIN
;  518        1460  3
;  519        1461  3           GET_FORM_SIZE (.SCB);                                 ! Returns the WidthxLength
;  520        1462  3
;  521        1463  3
;  522        1464  3           FORM_WIDTH      = .SCB[PSM$L_PAGE_WIDTH];
;  523        1465  3           FORM_LENGTH     = .SCB[PSM$L_PAGE_LENGTH];
;  524        1466  3
;  525        1467  3           RETURN_IF_ERROR_(ALLOCATE_PAGE( .SCB));              ! Get the page of memory
;  526        1468  3
;  527        1469  3           PAGE_REF = .SCB[PSM$A_PAGE_POINTER];                 ! My local page pointer
;  528        1470  3
;  529        1471  3           ! Always start at top of page
;  530        1472  3           !
;  531        1473  3           PAGE_REF[0,0,.FORM_WIDTH] = PSM$K_CHAR_FF;          ! form feed in 0 pos.
;  532        1474  3
;  533        1475  3           !  Standard Trailer Page 132x66:  text covers rows 1 through 58,
;  534        1476  3           !     translated to frames... ref starts at 1 and length is 57.
;  535        1477  3           FILL_FILE_TRAILER( .SCB,
;  536        1478  3                               PAGE_REF[0,1,.FORM_WIDTH],
;  537        1479  3                               .FORM_WIDTH,
;  538        1480  3                               .FORM_LENGTH - 6 - 2 - 1);   ! total form length...
;  539        1481  2           END;                                              ! ..6 burst, 2 spaces
;  540        1482  2                                                             ! ...top margin of 1
;  541        1483  2
;  542        1484  2       [PSM$K_CLOSE]:                   ! Return the Page of Memory
;  543        1485  2           RETURN_IF_ERROR_(DEALLOCATE_PAGE(.SCB));
;  544        1486  2
;  545        1487  2       [OTHERWISE]:
;  546        1488  2           RETURN PSM$_FUNNOTSUP;
;  547        1489  2
;  548        1490  2       TES; ! case .function
;  549        1491  2
;  550        1492  2   SS$_NORMAL
;  551        1493  2
;  552        1494  1   END;
```

```
                                    003C 00000          .ENTRY  PSM$FILE_TRAILER, Save R2,R3,R4,R5                    ; 1413
                     52        04   BC   D0 00002          MOVL    @SMB_CONTEXT, SCB                                     ; 1433
                     50        0C   BC   D0 00006          MOVL    @FUNCTION, R0                                         ; 1437
                     05             50   D1 0000A          CMPL    R0, #5                                                ; 1439
                                    41   12 0000D          BNEQ    2$                                                    ;
                     54      01FC   C2   D0 0000F          MOVL    508(SCB), PAGE_REF                                    ; 1443
          50    01F8 C2             08   C3 00014          SUBL3   #8, 504(SCB), R0                                      ; 1445
                     50      026C   C2   D1 0001A          CMPL    620(SCB), R0                                          ;
                                    08   15 0001F          BLEQ    1$                                                    ;
                     50 00000000G   8F   D0 00021          MOVL    #PSM$_EOF, R0                                         ; 1446
                                         04 00028          RET                                                           ;
                     53        10   AC   D0 00029 1$:      MOVL    FUNC_DESC, R3                                         ; 1448
                     63      0200   C2   D0 0002D          MOVL    512(SCB), (R3)                                        ;
          50    026C C2      0200   C2   C5 00032          MULL3   512(SCB), 620(SCB), R0                                ; 1450
     04   A3      50                54   C1 0003A          ADDL3   PAGE_REF, R0, 4(R3)                                   ;
                                    63   DD 0003F          PUSHL   (R3)                                                  ; 1454
                                    20   DD 00041          PUSHL   #32                                                   ; 1453
                               04   A3   DD 00043          PUSHL   4(R3)                                                 ;
              0000V  CF             03   FB 00046          CALLS   #3, DELIMIT_STRING_NOT                                ;
                     63             50   D0 0004B          MOVL    R0, (R3)                                              ;
                                    51   11 0004E          BRB     5$                                                    ; 1437
                     04             50   D1 00050 2$:      CMPL    R0, #4                                                ; 1458
                                    34   12 00053          BNEQ    3$                                                    ;
                                    52   DD 00055          PUSHL   SCB                                                   ; 1461
              0000V  CF             01   FB 00057          CALLS   #1, GET_FORM_SIZE                                     ;
                     55      0200   C2   D0 0005C          MOVL    512(SCB), FORM_WIDTH                                  ; 1464
                     53      01F8   C2   D0 00061          MOVL    504(SCB), FORM_LENGTH                                 ; 1465
                                    52   DD 00066          PUSHL   SCB                                                   ; 1467
              0000V  CF             01   FB 00068          CALLS   #1, ALLOCATE_PAGE                                     ;
                     34             50   E9 0006D          BLBC    STATUS, 6$                                            ;
                     54      01FC   C2   D0 00070          MOVL    508(SCB), PAGE_REF                                    ; 1469
                     64             0C   90 00075          MOVB    #12, (PAGE_REF)                                       ; 1473
                               F7   A3   9F 00078          PUSHAB  -9(FORM_LENGTH)                                       ; 1480
                                    55   DD 0007B          PUSHL   FORM_WIDTH                                            ; 1479
                                  6544   9F 0007D          PUSHAB  (FORM_WIDTH)[PAGE_REF]                                ; 1478
                                    52   DD 00080          PUSHL   SCB                                                   ;
              0000V  CF             04   FB 00082          CALLS   #4, FILL_FILE_TRAILER                                 ;
                                    18   11 00087          BRB     5$                                                    ; 1437
                     02             50   D1 00089 3$:      CMPL    R0, #2                                                ; 1484
                                    0B   12 0008C          BNEQ    4$                                                    ;
                                    52   DD 0008E          PUSHL   SCB                                                   ; 1485
              0000V  CF             01   FB 00090          CALLS   #1, DEALLOCATE_PAGE                                   ;
                     09             50   E8 00095          BLBS    STATUS, 5$                                            ;
                                         04 00098          RET                                                           ;
                     50 00000000G   8F   D0 00099 4$:      MOVL    #PSM$_FUNNOTSUP, R0                                   ; 1488
                                         04 000A0          RET                                                           ;
                     50             01   D0 000A1 5$:      MOVL    #1, R0                                                ; 1494
                                         04 000A4 6$:      RET                                                           ;

; Routine Size:  165 bytes,     Routine Base:  CODE + 0218
;  554          1495   1 %sbttl 'PSM$JOB_BURST - Print a Job Burst Page'
;  555          1496   1 ! Functional Description:
;  556          1497   1 !       This routine controls the creation of the job burst page.  The
;  557          1498   1 !       FUNCTION code dictates the action taken in creation.
;  558          1499   1 !       FUNCTION:
;  559          1500   1 !               OPEN  - Allocate and create the Job Burst Page
;  560          1501   1 !               READ  - Return the current line of the Job Burst Page
;  561          1502   1 !               CLOSE - Return the buffer allocated on OPEN
;  562          1503   1 !
;  563          1504   1 ! Formal Parameters:
;  564          1505   1 !               SMB_CONTEXT   - Pointer to the SMB
;  565          1506   1 !               USER_CONTEXT  - User defined pointer (not used here)
;  566          1507   1 !               FUNCTION      - OPEN, READ, CLOSE
;  567          1508   1 !               FUNC_DESC     - Pointer to functionally dependent descriptor
;  568          1509   1 !               FUNC_ARG      - Pointer to functionally dependent argument
;  569          1510   1 !
;  570          1511   1 ! Implicit Inputs:
;  571          1512   1 !               none
;  572          1513   1 !
;  573          1514   1 ! Implicit Outputs:
;  574          1515   1 !               none
;  575          1516   1 !
;  576          1517   1 ! Returned Value:
;  577          1518   1 !               none
;  578          1519   1 !
;  579          1520   1 ! Side Effects:
;  580          1521   1 !               none
;  581          1522   1 !--
;  582          1523   1 GLOBAL ROUTINE PSM$JOB_BURST ( %SBTTL 'JOB_BURST'
;  583          1524   1         SMB_CONTEXT     : REF VECTOR,
;  584          1525   1         USER_CONTEXT    : REF VECTOR,
;  585          1526   1         FUNCTION        : REF VECTOR,
;  586          1527   1         FUNC_DESC       : REF VECTOR,
;  587          1528   1         FUNC_ARG        : REF VECTOR
;  588          1529   1         )     =
;  589          1530   2 BEGIN
;  590          1531   2
;  591          1532   2 LOCAL
;  592          1533   2         SCB             : REF $BBLOCK,
;  593          1534   2         STATUS,
;  594          1535   2         FORM_WIDTH,
;  595          1536   2         FORM_LENGTH,
;  596          1537   2         FORM_SIZE,
;  597          1538   2         PAGE_REF        : REF PAGE_ARRAY;       ! Declare the pointer
;  598          1539   2                                                 !  to page
;  599          1540   2
;  600          1541   2 SCB = .SMB_CONTEXT[0];
;  601          1542   2
;  602          1543   2 ! Check the FUNCTION requested
;  603          1544   2
;  604          1545   2 SELECTONEU .FUNCTION[0] OF
;  605          1546   2 SET
;  606          1547   2     [PSM$K_READ]:
;  607          1548   3         BEGIN
;  608          1549   3         PAGE_REF = .SCB[PSM$A_PAGE_POINTER];
;  609          1550   3                                                 !  Output one line at a time
;  610          1551   4         IF .SCB[PSM$L_RECORD_NUMBER] GTR (.SCB[PSM$L_PAGE_LENGTH] - 6)
;  611          1552  3             THEN
;  612          1553  3                 RETURN PSM$_EOF;
;  613          1554  3
;  614          1555  3             FUNC_DESC[SIZE] = .SCB[PSM$L_PAGE_WIDTH];
;  615          1556  3             FUNC_DESC[ADDR] = PAGE_REF[0,.SCB[PSM$L_RECORD_NUMBER],
;  616          1557  3                                         .SCB[PSM$L_PAGE_WIDTH]];
;  617          1558  3
;  618          1559  3                                       ! adjust pointer
;  619          1560  3             FUNC_DESC[SIZE] = DELIMIT_STRING_NOT (.FUNC_DESC[ADDR],
;  620          1561  3                                                   %CHAR(32), .FUNC_DESC[SIZE]);
;  621          1562  3
;  622          1563  2             END;
;  623          1564  2
;  624          1565  2         [PSM$K_OPEN]:
;  625          1566  3             BEGIN
;  626          1567  3
;  627          1568  3             GET_FORM_SIZE (.SCB);                                ! Returns the WidthxLength
;  628          1569  3
;  629          1570  3
;  630          1571  3             FORM_WIDTH      = .SCB[PSM$L_PAGE_WIDTH];
;  631          1572  3             FORM_LENGTH     = .SCB[PSM$L_PAGE_LENGTH];
;  632          1573  3
;  633          1574  3             RETURN_IF_ERROR_(ALLOCATE_PAGE( .SCB)); ! Get the page of memory
;  634          1575  3
;  635          1576  3             PAGE_REF = .SCB[PSM$A_PAGE_POINTER];     ! My local page pointer
;  636          1577  3
;  637          1578  3             ! No form_feed on a burst page
;  638          1579  3
;  639          1580  3             !  Standard Burst Page 132x66:  text covers rows 2 through 60,
;  640          1581  3             !     translated to frames... ref starts at 2 and length is 58.
;  641          1582  3             FILL_JOB_FLAG(   .SCB,
;  642          1583  3                              PAGE_REF[0,2,.FORM_WIDTH],
;  643          1584  3                              .FORM_WIDTH,
;  644          1585  3                              .FORM_LENGTH - 6 - 2 );             ! 6 blank lines
;  645          1586  2             END;                                               ! top margin is 2
;  646          1587  2
;  647          1588  2         [PSM$K_CLOSE]:                    ! Return the Page of Memory
;  648          1589  2             RETURN_IF_ERROR_(DEALLOCATE_PAGE(.SCB));
;  649          1590  2
;  650          1591  2         [OTHERWISE]:
;  651          1592  2             RETURN PSM$_FUNNOTSUP;
;  652          1593  2
;  653          1594  2         TES; ! case .function
;  654          1595  2
;  655          1596  2 SS$_NORMAL
;  656          1597  2
;  657          1598  1 END;
```

```
                                        003C 00000        .ENTRY  PSM$JOB_BURST, Save R2,R3,R4,R5                     : 1523
                     52       04    BC   D0 00002        MOVL    @SMB_CONTEXT, SCB                                   : 1541
                     50       0C    BC   D0 00006        MOVL    @FUNCTION, R0                                       : 1545
                     05             50   D1 0000A        CMPL    R0, #5                                              : 1547
                                    41   12 0000D        BNEQ    2$                                                  :
```

```
                 55        01FC  C2  D0 0000F           MOVL     508(SCB), PAGE_REF                  ; 1549
     50    01F8  C2                  06  C3 00014           SUBL3    #6, 504(SCB), R0                    ; 1551
                 50        026C  C2  D1 0001A           CMPL     620(SCB), R0                        ;
                                 08  15 0001F           BLEQ     1$                                  ;
                 50   00000000G  8F  D0 00021           MOVL     #PSM$_EOF, R0                       ; 1553
                                 04     00028           RET                                          ;
                 53          10  AC  D0 00029 1$:       MOVL     FUNC_DESC, R3                       ; 1555
                 63        0200  C2  D0 0002D           MOVL     512(SCB), (R3)                      ;
     50    026C  C2        0200  C2  C5 00032           MULL3    512(SCB), 620(SCB), R0              ; 1557
 04  A3          50              55  C1 0003A           ADDL3    PAGE_REF, R0, 4(R3)                 ;
                                 63  DD 0003F           PUSHL    (R3)                                ; 1561
                                 20  DD 00041           PUSHL    #32                                 ; 1560
                             04  A3  DD 00043           PUSHL    4(R3)                               ;
          0000V  CF              03  FB 00046           CALLS    #3, DELIMIT_STRING_NOT              ;
                 63              50  D0 0004B           MOVL     R0, (R3)                            ;
                                 4E  11 0004E           BRB      5$                                  ; 1545
                 04              50  D1 00050 2$:       CMPL     R0, #4                              ; 1565
                                 31  12 00053           BNEQ     3$                                  ;
                                 52  DD 00055           PUSHL    SCB                                 ; 1568
          0000V  CF              01  FB 00057           CALLS    #1, GET_FORM_SIZE                   ;
                 53        0200  C2  D0 0005C           MOVL     512(SCB), FORM_WIDTH                ; 1571
                 54        01F8  C2  D0 00061           MOVL     504(SCB), FORM_LENGTH               ; 1572
                                 52  DD 00066           PUSHL    SCB                                 ; 1574
          0000V  CF              01  FB 00068           CALLS    #1, ALLOCATE_PAGE                   ;
                 31              50  E9 0006D           BLBC     STATUS, 6$                          ;
                 55        01FC  C2  D0 00070           MOVL     508(SCB), PAGE_REF                  ; 1576
                             F8  A4  9F 00075           PUSHAB   -8(FORM_LENGTH)                     ; 1585
                                 53  DD 00078           PUSHL    FORM_WIDTH                          ; 1584
                               6543  3F 0007A           PUSHAW   (PAGE_REF)[FORM_WIDTH]              ; 1583
                                 52  DD 0007D           PUSHL    SCB                                 ;
          0000V  CF              04  FB 0007F           CALLS    #4, FILL_JOB_FLAG                   ;
                                 18  11 00084           BRB      5$                                  ; 1545
                 02              50  D1 00086 3$:       CMPL     R0, #2                              ; 1588
                                 0B  12 00089           BNEQ     4$                                  ;
                                 52  DD 0008B           PUSHL    SCB                                 ; 1589
          0000V  CF              01  FB 0008D           CALLS    #1, DEALLOCATE_PAGE                 ;
                 09              50  E8 00092           BLBS     STATUS, 5$                          ;
                                 04     00095           RET                                          ;
                 50   00000000G  8F  D0 00096 4$:       MOVL     #PSM$_FUNNOTSUP, R0                 ; 1592
                                 04     0009D           RET                                          ;
                 50              01  D0 0009E 5$:       MOVL     #1, R0                              ; 1598
                                 04     000A1 6$:       RET                                          ;

; Routine Size:  162 bytes,    Routine Base:  CODE + 02BD
```

```
1 %sbttl 'PSM$JOB_FLAG - Print a Job Flag Page'
1 ! Functional Description:
1 !       This routine controls the creation of the job flag page.  The
1 !       FUNCTION code dictates the action taken in creation.
1 !       FUNCTION:
1 !               OPEN  - Allocate and create the Job Flag Page
1 !               READ  - Return the current line of the Job Flag Page
1 !               CLOSE - Return the buffer allocated on OPEN
1 !
1 ! Formal Parameters:
1 !               SMB_CONTEXT   - Pointer to the SMB
1 !               USER_CONTEXT  - User defined pointer (not used here)
1 !               FUNCTION      - OPEN, READ, CLOSE
1 !               FUNC_DESC     - Pointer to functionally dependent descriptor
1 !               FUNC_ARG      - Pointer to functionally dependent argument
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1 GLOBAL ROUTINE PSM$JOB_FLAG ( %SBTTL 'JOB_FLAG'
1               SMB_CONTEXT     : REF VECTOR,
1               USER_CONTEXT    : REF VECTOR,
1               FUNCTION        : REF VECTOR,
1               FUNC_DESC       : REF VECTOR,
1               FUNC_ARG        : REF VECTOR
1               )       =
2 BEGIN
2 LITERAL
2               TRAILING = 1;
2
2 LOCAL
2               SCB             : REF $BBLOCK,
2               STATUS,
2               FORM_WIDTH,
2               FORM_LENGTH,
2               FORM_SIZE,
2               BUFFER          : VECTOR [512,byte],    ! Assume max size 512 bytes
2               STRING_DESC     : VECTOR [2],           ! Descriptor to current string
2               PAGE_REF        : REF PAGE_ARRAY;       ! Declare the pointer
2                                                       !  to page
2
2 SCB = .SMB_CONTEXT[0];
2
2 ! Check the FUNCTION requested
2
2 SELECTONEU .FUNCTION[0] OF
2 SET
2     [PSM$K_READ]:
```

```
3            BEGIN
3            PAGE_REF = .SCB[PSM$A_PAGE_POINTER];
3                                                    !   Output one line at a time
4            IF (   .SCB[PSM$L_RECORD_NUMBER] GTR .SCB[PSM$L_PAGE_LENGTH])
3                            OR
4              ( NOT .SEPARATE_FLAG_(JOB_BURST) AND .SCB[PSM$L_RECORD_NUMBER]
4                            GEQ (.SCB[PSM$L_PAGE_LENGTH] - 6 ) )
3                    THEN
3                RETURN PSM$_EOF;
3
3            FUNC_DESC[SIZE] = .SCB[PSM$L_PAGE_WIDTH];
3            FUNC_DESC[ADDR] = PAGE_REF[0,.SCB[PSM$L_RECORD_NUMBER],
3                                       .SCB[PSM$L_PAGE_WIDTH]];
3
3                                     ! adjust pointer
3            FUNC_DESC[SIZE] = DELIMIT_STRING_NOT (.FUNC_DESC[ADDR],
3                                                  %CHAR(32), .FUNC_DESC[SIZE]);
3
2            END;
2
2        [PSM$K_OPEN]:
3            BEGIN
3
3            GET_FORM_SIZE (.SCB);                           ! Returns the WidthxLength
3
3
3            FORM_WIDTH       = .SCB[PSM$L_PAGE_WIDTH];
3            FORM_LENGTH      = .SCB[PSM$L_PAGE_LENGTH];
3
3            RETURN_IF_ERROR_(ALLOCATE_PAGE( .SCB));         ! Get the page of memory
3
3            PAGE_REF = .SCB[PSM$A_PAGE_POINTER];            ! My local page pointer
3
3            ! Always start at top of page
3            !
3            PAGE_REF[0,0,.FORM_WIDTH] = PSM$K_CHAR_FF;      ! form feed in 0 pos.
3
3            !  Standard Flag Page 132x66:  text covers rows 1 through 58,
3            !     translated to frames... ref starts at 1 and length is 57.
3            FILL_JOB_FLAG(   .SCB,
3                            PAGE_REF[0,1,.FORM_WIDTH],
3                            .FORM_WIDTH,
3                            .FORM_LENGTH - 6 - 2 - 1);       ! 6 burst,
3                                                            ! 2 spaces before burst,
3                                                            ! top margin is 1
4            IF (.SEPARATE_FLAG_(JOB_BURST))
3            THEN
4                BEGIN
4                STRING_DESC[SIZE] = %ALLOCATION(BUFFER);
4                STRING_DESC[ADDR] = BUFFER;
4
4                GET_VMS_LOGO
4                    (.SCB,
4                    STRING_DESC[0],                 ! Buffer descriptor
4                    STRING_DESC[SIZE]);             ! Returned length
4
4                INSERT_FRAME
```

```
;  773       1713  4                            (.SCB,
;  774       1714  4                            STRING_DESC[0],
;  775       1715  4                            PAGE_REF[10,.FORM_LENGTH-5,.FORM_WIDTH],
;  776       1716  4                            .FORM_WIDTH-20, 1);
;  777       1717  4                INSERT_FRAME
;  778       1718  4                            (.SCB,
;  779       1719  4                            STRING_DESC[0],
;  780       1720  4                            PAGE_REF[14,.FORM_LENGTH-4,.FORM_WIDTH],
;  781       1721  4                            .FORM_WIDTH-16, 1);
;  782       1722  4                INSERT_FRAME
;  783       1723  4                            (.SCB,
;  784       1724  4                            STRING_DESC[0],
;  785       1725  4                            PAGE_REF[10,.FORM_LENGTH-3,.FORM_WIDTH],
;  786       1726  4                            .FORM_WIDTH-20, 1);
;  787       1727  4                INSERT_FRAME
;  788       1728  4                            (.SCB,
;  789       1729  4                            STRING_DESC[0],
;  790       1730  4                            PAGE_REF[14,.FORM_LENGTH-2,.FORM_WIDTH],
;  791       1731  4                            .FORM_WIDTH-16, 1);
;  792       1732  4                INSERT_FRAME
;  793       1733  4                            (.SCB,
;  794       1734  4                            STRING_DESC[0],
;  795       1735  4                            PAGE_REF[10,.FORM_LENGTH-1,.FORM_WIDTH],
;  796       1736  4                            .FORM_WIDTH-20, 1);
;  797       1737  4                INSERT_FRAME
;  798       1738  4                            (.SCB,
;  799       1739  4                            STRING_DESC[0],
;  800       1740  4                            PAGE_REF[14,.FORM_LENGTH,.FORM_WIDTH],
;  801       1741  4                            .FORM_WIDTH-16, 1);
;  802       1742  3                END;
;  803       1743  2            END;
;  804       1744  2
;  805       1745  2        [PSM$K_CLOSE]:                          ! Return the Page of Memory
;  806       1746  2            RETURN_IF_ERROR_(DEALLOCATE_PAGE(.SCB));
;  807       1747  2
;  808       1748  2        [OTHERWISE]:
;  809       1749  2            RETURN PSM$_FUNNOTSUP;
;  810       1750  2
;  811       1751  2        TES; ! case .function
;  812       1752  2
;  813       1753  2    SS$_NORMAL
;  814       1754  2
;  815       1755  1    END;



                                  007C 00000         .ENTRY  PSM$JOB_FLAG, Save R2,R3,R4,R5,R6               ; 1627
                56     0000V   CF  9E 00002         MOVAB   INSERT_FRAME, R6                                ;
                5E      FDF8   CE  9E 00007         MOVAB   -520(SP), SP                                    ;
                54        04   BC  D0 0000C         MOVL    @SMB_CONTEXT, SCB                               ; 1649
                50        0C   BC  D0 00010         MOVL    @FUNCTION, R0                                   ; 1653
                05             50  D1 00014         CMPL    R0, #5                                          ; 1655
                               4F  12 00017         BNEQ    4$                                              ;
                53      01FC   C4  D0 00019         MOVL    508(SCB), PAGE_REF                              ; 1657
                50      026C   C4  D0 0001E         MOVL    620(SCB), R0                                    ; 1659
```

```
              01F8    C4                50 D1 00023       CMPL    R0, 504(SCB)                            ;
                                        11 14 00028       BGTR    1$                                      ;
        13    0154    C4                05 E0 0002A       BBS     #5, 340(SCB), 2$                        ; 1661
        51    01F8    C4                06 C3 00030       SUBL3   #6, 504(SCB), R1                        ; 1662
                      51                50 D1 00036       CMPL    R0, R1                                  ;
                                        08 19 00039       BLSS    2$                                      ;
                      50   00000000G    8F D0 0003B 1$:   MOVL    #PSM$_EOF, R0                           ; 1664
                                           04 00042       RET                                             ;
                      52          10    AC D0 00043 2$:   MOVL    FUNC_DESC, R2                           ; 1666
                      62        0200    C4 D0 00047       MOVL    512(SCB), (R2)                          ;
                      50        0200    C4 C4 0004C       MULL2   512(SCB), R0                            ; 1668
  04    A2            50                53 C1 00051       ADDL3   PAGE_REF, R0, 4(R2)                     ;
                                        62 DD 00056       PUSHL   (R2)                                    ; 1672
                                        20 DD 00058       PUSHL   #32                                     ; 1671
                                  04    A2 DD 0005A       PUSHL   4(R2)                                   ;
             0000V    CF                03 FB 0005D       CALLS   #3, DELIMIT_STRING_NOT                  ;
                      62                50 D0 00062       MOVL    R0, (R2)                                ;
                                      00FD 31 00065 3$:   BRW     9$                                      ; 1653
                      04                50 D1 00068 4$:   CMPL    R0, #4                                  ; 1676
                                        03 13 0006B       BEQL    5$                                      ;
                                      00DD 31 0006D       BRW     7$                                      ;
                                        54 DD 00070 5$:   PUSHL   SCB                                     ; 1679
             0000V    CF                01 FB 00072       CALLS   #1, GET_FORM_SIZE                       ;
                      55        0200    C4 D0 00077       MOVL    512(SCB), FORM_WIDTH                    ; 1682
                      52        01F8    C4 D0 0007C       MOVL    504(SCB), FORM_LENGTH                   ; 1683
                                        54 DD 00081       PUSHL   SCB                                     ; 1685
             0000V    CF                01 FB 00083       CALLS   #1, ALLOCATE_PAGE                       ;
                      01                50 E8 00088       BLBS    STATUS, 6$                              ;
                                           04 0008B       RET                                             ;
                      53        01FC    C4 D0 0008C 6$:   MOVL    508(SCB), PAGE_REF                      ; 1687
                      63                0C 90 00091       MOVB    #12, (PAGE_REF)                         ; 1691
                                  F7    A2 9F 00094       PUSHAB  -9(FORM_LENGTH)                         ; 1698
                                        55 DD 00097       PUSHL   FORM_WIDTH                              ; 1697
                                      6543 9F 00099       PUSHAB  (FORM_WIDTH)[PAGE_REF]                  ; 1696
                                        54 DD 0009C       PUSHL   SCB                                     ;
             0000V    CF                04 FB 0009E       CALLS   #4, FILL_JOB_FLAG                       ;
        BC    0154    C4                05 E1 000A3       BBC     #5, 340(SCB), 3$                        ; 1701
                      6E        0200    8F 3C 000A9       MOVZWL  #512, STRING_DESC                       ; 1704
                04    AE          08    AE 9E 000AE       MOVAB   BUFFER, STRING_DESC+4                   ; 1705
                                        5E DD 000B3       PUSHL   SP                                      ; 1710
                                  04    AE 9F 000B5       PUSHAB  STRING_DESC                             ; 1709
                                        54 DD 000B8       PUSHL   SCB                                     ; 1708
             0000V    CF                03 FB 000BA       CALLS   #3, GET_VMS_LOGO                        ;
                                        01 DD 000BF       PUSHL   #1                                      ; 1715
                                  EC    A5 9F 000C1       PUSHAB  -20(FORM_WIDTH)                         ; 1716
                      50          FB    A2 9E 000C4       MOVAB   -5(R2), R0                              ; 1715
                      50                55 C4 000C8       MULL2   FORM_WIDTH, R0                          ;
                                  0A  A043 9F 000CB       PUSHAB  10(R0)[PAGE_REF]                        ;
                                  0C    AE 9F 000CF       PUSHAB  STRING_DESC                             ; 1714
                                        54 DD 000D2       PUSHL   SCB                                     ; 1715
                      66                05 FB 000D4       CALLS   #5, INSERT_FRAME                        ;
                                        01 DD 000D7       PUSHL   #1                                      ; 1720
                                  F0    A5 9F 000D9       PUSHAB  -16(FORM_WIDTH)                         ; 1721
                      50          FC    A2 9E 000DC       MOVAB   -4(R2), R0                              ; 1720
                      50                55 C4 000E0       MULL2   FORM_WIDTH, R0                          ;
                                  0E  A043 9F 000E3       PUSHAB  14(R0)[PAGE_REF]                        ;
                                  0C    AE 9F 000E7       PUSHAB  STRING_DESC                             ; 1719
```

```
                                    54  DD 000EA          PUSHL    SCB                                   ; 1720
                  66                05  FB 000EC          CALLS    #5, INSERT_FRAME                      ;
                                    01  DD 000EF          PUSHL    #1                                    ; 1725
                          EC        A5  9F 000F1          PUSHAB   -20(FORM_WIDTH)                       ; 1726
                  50      FD        A2  9E 000F4          MOVAB    -3(R2), R0                            ; 1725
                  50                55  C4 000F8          MULL2    FORM_WIDTH, R0                        ;
                          0A A043       9F 000FB          PUSHAB   10(R0)[PAGE_REF]                      ;
                          0C        AE  9F 000FF          PUSHAB   STRING_DESC                           ; 1724
                                    54  DD 00102          PUSHL    SCB                                   ; 1725
                  66                05  FB 00104          CALLS    #5, INSERT_FRAME                      ;
                                    01  DD 00107          PUSHL    #1                                    ; 1730
                          F0        A5  9F 00109          PUSHAB   -16(FORM_WIDTH)                       ; 1731
                  50      FE        A2  9E 0010C          MOVAB    -2(R2), R0                            ; 1730
                  50                55  C4 00110          MULL2    FORM_WIDTH, R0                        ;
                          0E A043       9F 00113          PUSHAB   14(R0)[PAGE_REF]                      ;
                          0C        AE  9F 00117          PUSHAB   STRING_DESC                           ; 1729
                                    54  DD 0011A          PUSHL    SCB                                   ; 1730
                  66                05  FB 0011C          CALLS    #5, INSERT_FRAME                      ;
                                    01  DD 0011F          PUSHL    #1                                    ; 1735
                          EC        A5  9F 00121          PUSHAB   -20(FORM_WIDTH)                       ; 1736
                  50      FF        A2  9E 00124          MOVAB    -1(R2), R0                            ; 1735
                  50                55  C4 00128          MULL2    FORM_WIDTH, R0                        ;
                          0A A043       9F 0012B          PUSHAB   10(R0)[PAGE_REF]                      ;
                          0C        AE  9F 0012F          PUSHAB   STRING_DESC                           ; 1734
                                    54  DD 00132          PUSHL    SCB                                   ; 1735
                  66                05  FB 00134          CALLS    #5, INSERT_FRAME                      ;
                                    01  DD 00137          PUSHL    #1                                    ; 1740
                          F0        A5  9F 00139          PUSHAB   -16(FORM_WIDTH)                       ; 1741
                  52                55  C4 0013C          MULL2    FORM_WIDTH, R2                        ; 1740
                          0E A243       9F 0013F          PUSHAB   14(R2)[PAGE_REF]                      ;
                          0C        AE  9F 00143          PUSHAB   STRING_DESC                           ; 1739
                                    54  DD 00146          PUSHL    SCB                                   ; 1740
                  66                05  FB 00148          CALLS    #5, INSERT_FRAME                      ;
                                    18  11 0014B          BRB      9$                                    ; 1653
                  02                50  D1 0014D 7$:      CMPL     R0, #2                                ; 1745
                                    0B  12 00150          BNEQ     8$                                    ;
                                    54  DD 00152          PUSHL    SCB                                   ; 1746
            0000V CF                01  FB 00154          CALLS    #1, DEALLOCATE_PAGE                   ;
                  09                50  E8 00159          BLBS     STATUS, 9$                            ;
                                        04 0015C          RET                                            ;
                  50 00000000G      8F  D0 0015D 8$:      MOVL     #PSM$_FUNNOTSUP, R0                   ; 1749
                                        04 00164          RET                                            ;
                  50                01  D0 00165 9$:      MOVL     #1, R0                                ; 1755
                                        04 00168          RET                                            ;

; Routine Size:  361 bytes,    Routine Base:  CODE + 035F
```

```
1 %sbttl 'PSM$JOB_TRAILER - Print a Job Flag Page'
1 ! Functional Description:
1 !         This routine controls the creation of the job trailer page.  The
1 !         FUNCTION code dictates the action taken in creation.
1 !         FUNCTION:
1 !                 OPEN  - Allocate and create the Job Trailer Page
1 !                 READ  - Return the current line of the Job Trailer Page
1 !                 CLOSE - Return the buffer allocated on OPEN
1 !
1 ! Formal Parameters:
1 !                 SMB_CONTEXT   - Pointer to the SMB
1 !                 USER_CONTEXT  - User defined pointer (not used here)
1 !                 FUNCTION      - OPEN, READ, CLOSE
1 !                 FUNC_DESC     - Pointer to functionally dependent descriptor
1 !                 FUNC_ARG      - Pointer to functionally dependent argument
1 !
1 ! Implicit Inputs:
1 !                 none
1 !
1 ! Implicit Outputs:
1 !                 none
1 !
1 ! Returned Value:
1 !                 none
1 !
1 ! Side Effects:
1 !                 none
1 !--
1 GLOBAL ROUTINE PSM$JOB_TRAILER ( %SBTTL 'JOB_TRAILER'
1         SMB_CONTEXT     : REF VECTOR,
1         USER_CONTEXT    : REF VECTOR,
1         FUNCTION        : REF VECTOR,
1         FUNC_DESC       : REF VECTOR,
1         FUNC_ARG        : REF VECTOR
1         ) =
2 BEGIN
2
2 LITERAL
2         TRAILING = 1;
2 LOCAL
2         SCB             : REF $BBLOCK,
2         STATUS,
2         FORM_WIDTH,
2         FORM_LENGTH,
2         FORM_SIZE,
2         PAGE_REF        : REF PAGE_ARRAY;       ! Declare the pointer
2                                                 !  to page
2
2 SCB = .SMB_CONTEXT[0];
2
2 ! Check the FUNCTION requested
2
2 SELECTONEU .FUNCTION[0] OF
2 SET
2     [PSM$K_READ]:
3         BEGIN
3         LOCAL  TEMP_PTR;
```

```
3
3         PAGE_REF = .SCB[PSM$A_PAGE_POINTER];
3
4         IF .SCB[PSM$L_RECORD_NUMBER] GTR (.SCB[PSM$L_PAGE_LENGTH] - 2 - 6)
3         THEN
3             RETURN PSM$_EOF;
3
3         FUNC_DESC[SIZE] = .SCB[PSM$L_PAGE_WIDTH];
3         FUNC_DESC[ADDR] = PAGE_REF[0,.SCB[PSM$L_RECORD_NUMBER],
3                                  .SCB[PSM$L_PAGE_WIDTH]];
3
3                                 ! adjust pointer
3         FUNC_DESC[SIZE] = DELIMIT_STRING_NOT (.FUNC_DESC[ADDR],
3                                               %CHAR(32), .FUNC_DESC[SIZE]);
3
2         END;
2
2     [PSM$K_OPEN]:
3         BEGIN
3
3         GET_FORM_SIZE (.SCB);                                   ! Returns the WidthxLength
3
3
3         FORM_WIDTH        = .SCB[PSM$L_PAGE_WIDTH];
3         FORM_LENGTH       = .SCB[PSM$L_PAGE_LENGTH];
3
3         RETURN_IF_ERROR_(ALLOCATE_PAGE( .SCB));                 ! Get the page of memory
3
3         PAGE_REF = .SCB[PSM$A_PAGE_POINTER];                    ! My local page pointer
3
3         ! Always start at top of page
3         !
3         PAGE_REF[0,0,.FORM_WIDTH] = PSM$K_CHAR_FF;              ! form feed in 0 pos.
3
3         !  Standard Trailer Page 132x66:  text covers rows 1 through 58,
3         !     translated to frames... ref starts at 1 and length is 57.
3         FILL_JOB_TRAILER( .SCB,
3                           PAGE_REF[0,1,.FORM_WIDTH],
3                           .FORM_WIDTH,
3                           .FORM_LENGTH - 6 - 2 - 1 );           ! ...6 burst, 2 sp,
2         END;                                                    !     top margin is 1
2
2     [PSM$K_CLOSE]:                          ! Return the Page of Memory
2         RETURN_IF_ERROR_(DEALLOCATE_PAGE(.SCB));
2
2     [OTHERWISE]:
2         RETURN PSM$_FUNNOTSUP;
2
2     TES; ! case .function
2
2 SS$_NORMAL
2
1 END;
```

```
                                         003C 00000         .ENTRY   PSM$JOB_TRAILER, Save R2,R3,R4,R5             ; 1784
                   52               04  BC  D0 00002         MOVL     @SMB_CONTEXT, SCB                              ; 1804
                   50               0C  BC  D0 00006         MOVL     @FUNCTION, R0                                  ; 1808
                   05                   50  D1 0000A         CMPL     R0, #5                                         ; 1810
                                        41  12 0000D         BNEQ     2$                                             ;
                   54             01FC  C2  D0 0000F         MOVL     508(SCB), PAGE_REF                             ; 1814
       50   01F8   C2                   08  C3 00014         SUBL3    #8, 504(SCB), R0                               ; 1816
                   50             026C  C2  D1 0001A         CMPL     620(SCB), R0                                   ;
                                        08  15 0001F         BLEQ     1$                                             ;
                   50 00000000G         8F  D0 00021         MOVL     #PSM$_EOF, R0                                  ; 1818
                                            04 00028         RET                                                     ;
                   53               10  AC  D0 00029 1$:     MOVL     FUNC_DESC, R3                                  ; 1820
                   63             0200  C2  D0 0002D         MOVL     512(SCB), (R3)                                 ;
       50   026C   C2             0200  C2  C5 00032         MULL3    512(SCB), 620(SCB), R0                         ; 1822
04     A3          50                   54  C1 0003A         ADDL3    PAGE_REF, R0, 4(R3)                            ;
                                        63  DD 0003F         PUSHL    (R3)                                           ; 1826
                                        20  DD 00041         PUSHL    #32                                            ; 1825
                                    04  A3  DD 00043         PUSHL    4(R3)                                          ;
            0000V  CF                   03  FB 00046         CALLS    #3, DELIMIT_STRING_NOT                         ;
                   63                   50  D0 0004B         MOVL     R0, (R3)                                       ;
                                        51  11 0004E         BRB      5$                                             ; 1808
                   04                   50  D1 00050 2$:     CMPL     R0, #4                                         ; 1830
                                        34  12 00053         BNEQ     3$                                             ;
                                        52  DD 00055         PUSHL    SCB                                            ; 1833
            0000V  CF                   01  FB 00057         CALLS    #1, GET_FORM_SIZE                              ;
                   55             0200  C2  D0 0005C         MOVL     512(SCB), FORM_WIDTH                           ; 1836
                   53             01F8  C2  D0 00061         MOVL     504(SCB), FORM_LENGTH                          ; 1837
                                        52  DD 00066         PUSHL    SCB                                            ; 1839
            0000V  CF                   01  FB 00068         CALLS    #1, ALLOCATE_PAGE                              ;
                   34                   50  E9 0006D         BLBC     STATUS, 6$                                     ;
                   54             01FC  C2  D0 00070         MOVL     508(SCB), PAGE_REF                             ; 1841
                   64                   0C  90 00075         MOVB     #12, (PAGE_REF)                                ; 1845
                                    F7  A3  9F 00078         PUSHAB   -9(FORM_LENGTH)                                ; 1852
                                        55  DD 0007B         PUSHL    FORM_WIDTH                                     ; 1851
                                      6544  9F 0007D         PUSHAB   (FORM_WIDTH)[PAGE_REF]                         ; 1850
                                        52  DD 00080         PUSHL    SCB                                            ;
            0000V  CF                   04  FB 00082         CALLS    #4, FILL_JOB_TRAILER                           ;
                                        18  11 00087         BRB      5$                                             ; 1808
                   02                   50  D1 00089 3$:     CMPL     R0, #2                                         ; 1855
                                        0B  12 0008C         BNEQ     4$                                             ;
                                        52  DD 0008E         PUSHL    SCB                                            ; 1856
            0000V  CF                   01  FB 00090         CALLS    #1, DEALLOCATE_PAGE                            ;
                   09                   50  E8 00095         BLBS     STATUS, 5$                                     ;
                                            04 00098         RET                                                     ;
                   50 00000000G         8F  D0 00099 4$:     MOVL     #PSM$_FUNNOTSUP, R0                            ; 1859
                                            04 000A0         RET                                                     ;
                   50                   01  D0 000A1 5$:     MOVL     #1, R0                                         ; 1865
                                            04 000A4 6$:     RET                                                     ;

; Routine Size:  165 bytes,    Routine Base:  CODE + 04C8
```

```
1 %sbttl 'PSM$PAGE_HEADER - Print a Header at the Top of each Page'
1 ! Functional Description:
1 !       Creates a page header for the current file and prints it at the
1 !       top of each page.
1 !       FUNCTION:
1 !               OPEN  - Allocate and create the Page Header
1 !               READ  - Return the current header with the new page number
1 !               CLOSE - Deallocate the header
1 !
1 ! Formal Parameters:
1 !               SMB_CONTEXT   - Pointer to the SMB
1 !               USER_CONTEXT  - User defined pointer (not used here)
1 !               FUNCTION      - OPEN, READ, CLOSE
1 !               FUNC_DESC     - Pointer to functionally dependent descriptor
1 !               FUNC_ARG      - Pointer to functionally dependent argument
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1 GLOBAL ROUTINE PSM$PAGE_HEADER ( %SBTTL 'PAGE_HEADER'
1       SMB_CONTEXT     : REF VECTOR,
1       USER_CONTEXT    : REF VECTOR,
1       FUNCTION        : REF VECTOR,
1       FUNC_DESC       : REF VECTOR,
1       FUNC_ARG        : REF VECTOR
1       ) =
2 BEGIN
2 LOCAL
2       SCB             : REF $BBLOCK;
2
2 SCB = .SMB_CONTEXT[0];
2
2 ! Check the FUNCTION requested
2
2 SELECTONEU .FUNCTION[0] OF
2 SET
2     [PSM$K_READ]:
3       BEGIN
3
3       IF .SCB[PSM$L_RECORD_NUMBER] GTRU 0
3       THEN
3           RETURN PSM$_EOF;
3
3       ! Use the supplied string descriptor as a temp for the page number
3       !
3       FUNC_DESC[SIZE] = 5;
3       FUNC_DESC[ADDR] = .SCB_ADDR_(PAGE_HEADER) + .SCB_SIZE_(PAGE_HEADER) - 8;
3       !
```

```
            ! Write the page number into the end of the page header buffer
            ! (note -- since the page number can decrease we always fill out
            ! the page number area with blanks to overwrite any prior data)
            !
            $FAO (
                $DESCRIPTOR ('!5<!UL!>' ),              ! pad with trailing spaces
                FUNC_DESC[SIZE],                        ! ignore return length
                FUNC_DESC[0],                           ! temp output buffer desc
                .SCB[PSM$L_PAGE]                        ! current page number
                );

            ! copy the page header descriptor size and address to
            ! the function descriptor
            !
            FUNC_DESC[SIZE] = .SCB_SIZE_ (PAGE_HEADER);
            FUNC_DESC[ADDR] = .SCB_ADDR_ (PAGE_HEADER);

            END;

        [PSM$K_OPEN]:
            BEGIN
            !  set carriage control to imbedded
            !
            FUNC_ARG[0] = PSM$K_CC_INTERNAL;

            ! Format everything but the page number, but only do it once per task
            !
            IF TESTBITCS (SCB[PSM$V_PAGE_HEADER_BUILT])
            THEN
                CREATE_PAGE_HEADER (.SCB);

            END;

        [PSM$K_START_TASK]:
            ! Set the size of the page header equal to the page width adjusted
            ! for margins.
            !
            BEGIN
            GET_FORM_SIZE (.SCB);                       ! Returns the WidthxLength

            ! Adjust for margins and imbedded carriage control
            !
            SCB[PSM$L_PAGE_WIDTH] = .SCB[PSM$L_PAGE_WIDTH]
                            - .SCB[PSM$L_LEFT_MARGIN]       ! less leading spaces
                            - .SCB[PSM$L_RIGHT_MARGIN]      ! less early truncation
                            + 3;                            ! plus trailing carr
                                                            ! cntrl <CR><LF><LF>

            RETURN_IF_ERROR_ (STR$GET1_DX (%REF (.SCB[PSM$L_PAGE_WIDTH]),
                                SCB[PSM$Q_PAGE_HEADER]));
            END;

        [OTHERWISE]:
            RETURN PSM$_FUNNOTSUP;

        TES; ! case .function
```

```
; 1042        1980  2 SS$_NORMAL
; 1043        1981  2
; 1044        1982  1 END;


                    3E  21  4C  55  21  3C  35  21  0056D P.AAB:  .ASCII   \!5<!UL!>\                             ;
                                                    00575         .BLKB    3
                                         00000008  00578 P.AAA:  .LONG    8                                      ;
                                         00000000' 0057C         .ADDRESS P.AAB

                                                                 .EXTRN   SYS$FAO

                                              001C 00000         .ENTRY   PSM$PAGE_HEADER, Save R2,R3,R4         ; 1894
                5E                        04   C2 00002         SUBL2    #4, SP                                 ;
                52          04            BC   D0 00005         MOVL     @SMB_CONTEXT, SCB                      ; 1905
                50          0C            BC   D0 00009         MOVL     @FUNCTION, R0                          ; 1909
                05                        50   D1 0000D         CMPL     R0, #5                                 ; 1911
                                          46   12 00010         BNEQ     2$                                     ;
                          026C            C2   D5 00012         TSTL     620(SCB)                               ; 1914
                                          08   13 00016         BEQL     1$                                     ;
                50   00000000G            8F   D0 00018         MOVL     #PSM$_EOF, R0                          ; 1916
                                               04 0001F         RET                                             ;
                53          10            AC   D0 00020 1$:     MOVL     FUNC_DESC, R3                          ; 1920
                63                        05   D0 00024         MOVL     #5, (R3)                               ;
                54                        52   D0 00027         MOVL     SCB, R4                                ; 1921
                50        01F0            C4   3C 0002A         MOVZWL   496(R4), R0                            ;
                50        01F4            C2   C0 0002F         ADDL2    500(SCB), R0                           ;
          04    A3          F8            A0   9E 00034         MOVAB    -8(R0), 4(R3)                          ;
                          01EC            C2   DD 00039         PUSHL    492(SCB)                               ; 1932
                                          53   DD 0003D         PUSHL    R3                                     ;
                                          53   DD 0003F         PUSHL    R3                                     ;
                            B4            AF   9F 00041         PUSHAB   P.AAA                                  ;
   00000000G    00                        04   FB 00044         CALLS    #4, SYS$FAO                            ;
                63        01F0            C4   3C 0004B         MOVZWL   496(R4), (R3)                          ; 1937
          04    A3        01F4            C2   D0 00050         MOVL     500(SCB), 4(R3)                        ; 1938
                                          55   11 00056         BRB      5$                                     ; 1909
                04                        50   D1 00058 2$:     CMPL     R0, #4                                 ; 1942
                                          12   12 0005B         BNEQ     3$                                     ;
          14    BC                        01   D0 0005D         MOVL     #1, @FUNC_ARG                          ; 1946
   47     10    A2                        08   E2 00061         BBSS     #8, 16(SCB), 5$                        ; 1950
                                          52   DD 00066         PUSHL    SCB                                    ; 1952
       0000V    CF                        01   FB 00068         CALLS    #1, CREATE_PAGE_HEADER                 ;
                                          3E   11 0006D         BRB      5$                                     ; 1909
                10                        50   D1 0006F 3$:     CMPL     R0, #16                                ; 1956
                                          31   12 00072         BNEQ     4$                                     ;
                                          52   DD 00074         PUSHL    SCB                                    ; 1961
       0000V    CF                        01   FB 00076         CALLS    #1, GET_FORM_SIZE                      ;
                51        0200            C2   9E 0007B         MOVAB    512(SCB), R1                           ; 1965
   50           61        00BC            C2   C3 00080         SUBL3    188(SCB), (R1), R0                     ; 1966
                50        0148            C2   C2 00086         SUBL2    328(SCB), R0                           ; 1967
                61          03            A0   9E 0008B         MOVAB    3(R0), (R1)                            ; 1968
                          01F0            C2   9F 0008F         PUSHAB   496(SCB)                               ; 1972
          04    AE                        61   D0 00093         MOVL     (R1), 4(SP)                            ;
                            04            AE   9F 00097         PUSHAB   4(SP)                                  ;
   00000000G    00                        02   FB 0009A         CALLS    #2, STR$GET1_DX                        ;
                09                        50   E8 000A1         BLBS     STATUS, 5$                             ;
```

```
                           04 000A4        RET                                       ;
   50 00000000G   8F       D0 000A5 4$:    MOVL    #PSM$_FUNNOTSUP, R0               ; 1976
                           04 000AC        RET                                       ;
   50             01       D0 000AD 5$:    MOVL    #1, R0                            ; 1982
                           04 000B0        RET                                       ;

; Routine Size:  177 bytes,    Routine Base:  CODE + 0580
; 1046          1983 1 ROUTINE PARSE_FILE_NAME ( %SBTTL 'PARSE_FILE_NAME'
; 1047          1984 1          FILENAME         : REF $BBLOCK,
; 1048          1985 1          ITEM_CODE        ,
; 1049          1986 1          RESULT           : REF VECTOR
; 1050          1987 1          )        =
; 1051          1988 2 BEGIN
; 1052          1989 2
; 1053          1990 2 LOCAL
; 1054          1991 2          LIST     : $ITMBLK [1,8]
; 1055          1992 2          ;
; 1056          1993 2
; 1057          1994 2 CH$FILL (0, %ALLOCATION (LIST), LIST);
; 1058          1995 2
; 1059          1996 2 LIST [0, ITM$W_ITMCOD] = .ITEM_CODE;
; 1060          1997 2
; 1061          1998 2 RETURN_IF_ERROR_ ($FILESCAN (SRCSTR=.FILENAME, VALUELST=LIST));
; 1062          1999 2
; 1063          2000 2 RESULT[SIZE] = .LIST[0, ITM$W_BUFSIZ];
; 1064          2001 2 RESULT[ADDR] = .LIST[0, ITM$L_BUFADR];
; 1065          2002 2
; 1066          2003 2 SS$_NORMAL
; 1067          2004 2
; 1068          2005 1 END;



                                                                         .EXTRN  SYS$FILESCAN

                                                     003C 00000 PARSE_FILE_NAME:
                                                                         .WORD   Save R2,R3,R4,R5                             ; 1983
                                    5E           0C  C2 00002           SUBL2   #12, SP                                      ;
          0C            00          6E           00  2C 00005           MOVC5   #0, (SP), #0, #12, LIST                      ; 1994
                                                 6E     0000A                                                                ;
                           02       AE     08    AC  B0 0000B           MOVW    ITEM_CODE, LIST+2                            ; 1996
                                                 7E  D4 00010           CLRL    -(SP)                                        ; 1998
                                           04    AE  9F 00012           PUSHAB  LIST                                         ;
                                           04    AC  DD 00015           PUSHL   FILENAME                                     ;
                       00000000G    00           03  FB 00018           CALLS   #3, SYS$FILESCAN                             ;
                                    0F           50  E9 0001F           BLBC    STATUS, 1$                                   ;
                                    50     0C    AC  D0 00022           MOVL    RESULT, R0                                   ; 2000
                                    60           6E  3C 00026           MOVZWL  LIST, (R0)                                   ;
                           04       A0     04    AE  D0 00029           MOVL    LIST+4, 4(R0)                                ; 2001
                                    50           01  D0 0002E           MOVL    #1, R0                                       ; 2005
                                                     04 00031 1$:       RET                                                  ;

; Routine Size:  50 bytes,    Routine Base:  CODE + 0631
; 1070          2006  1 %sbttl 'ALLOCATE_PAGE - Allocate the Page of Memory'
; 1071          2007  1 !++
; 1072          2008  1 ! Functional Description:
; 1073          2009  1 !       This routine allocates memory in an amount of
; 1074          2010  1 !       memory equal to the largest Form Size supported.
; 1075          2011  1 !
; 1076          2012  1 ! Formal Parameters:
; 1077          2013  1 !               SCB     - Address of the SCB
; 1078          2014  1 !
; 1079          2015  1 ! Implicit Inputs:
; 1080          2016  1 !               none
; 1081          2017  1 !
; 1082          2018  1 ! Implicit Outputs:
; 1083          2019  1 !               none
; 1084          2020  1 !
; 1085          2021  1 ! Returned Value:
; 1086          2022  1 !               none
; 1087          2023  1 !
; 1088          2024  1 ! Side Effects:
; 1089          2025  1 !               none
; 1090          2026  1 !--
; 1091          2027  1 ROUTINE ALLOCATE_PAGE(
; 1092          2028  1                 SCB   : REF $BBLOCK
; 1093          2029  1                     ) =
; 1094          2030  2 BEGIN
; 1095          2031  2    LOCAL
; 1096          2032  2         PAGE_SIZE;
; 1097          2033  2
; 1098          2034  2     PAGE_SIZE = .SCB[PSM$L_PAGE_WIDTH] * (.SCB[PSM$L_PAGE_LENGTH]+1);
; 1099          2035  2
; 1100        P 2036  2     RETURN_IF_ERROR_( LIB$GET_VM ( %REF(.PAGE_SIZE),
; 1101          2037  2                       SCB[PSM$A_PAGE_POINTER]));
; 1102          2038  2                                                              ! Fill it with Blanks
; 1103          2039  2     CH$FILL (%CHAR(32), .PAGE_SIZE,
; 1104          2040  2             .SCB[PSM$A_PAGE_POINTER]);
; 1105          2041  2
; 1106          2042  2     RETURN SS$_NORMAL;
; 1107          2043  1 END;



                                       003C 00000 ALLOCATE_PAGE:
                                                         .WORD    Save R2,R3,R4,R5                              ; 2027
                   5E          04  C2 00002              SUBL2    #4, SP                                        :
                   53     04   AC  D0 00005              MOVL     SCB, R3                                       ; 2034
        50    01F8 C3          01  C1 00009              ADDL3    #1, 504(R3), R0                               :
        52         50   0200   C3  C5 0000F              MULL3    512(R3), R0, PAGE_SIZE                        :
                        01FC   C3  9F 00015              PUSHAB   508(R3)                                       ; 2037
             04    AE          52  D0 00019              MOVL     PAGE_SIZE, 4(SP)                              :
                          04   AE  9F 0001D              PUSHAB   4(SP)                                         :
        00000000G  00          02  FB 00020              CALLS    #2, LIB$GET_VM                                :
                   0B          50  E9 00027              BLBC     STATUS, 1$                                    :
  52         20    6E          00  2C 0002A              MOVC5    #0, (SP), #32, PAGE_SIZE, @508(R3)            ; 2040
                        01FC   D3     0002F                                                                     :
                   50          01  D0 00032              MOVL     #1, R0                                        ; 2042
```

```
                                        04 00035 1$:    RET                                        ; 2043

; Routine Size:  54 bytes,    Routine Base:  CODE + 0663
; 1109          2044  1 %sbttl 'DEALLOCATE_PAGE - Deallocate the Page of Memory'
; 1110          2045  1 !++
; 1111          2046  1 ! Functional Description:
; 1112          2047  1 !       This routine deallocates memory in an amount of
; 1113          2048  1 !       memory equal to the largest Form Size supported.
; 1114          2049  1 !
; 1115          2050  1 ! Formal Parameters:
; 1116          2051  1 !               SCB     - Address of the SCB
; 1117          2052  1 !
; 1118          2053  1 ! Implicit Inputs:
; 1119          2054  1 !               none
; 1120          2055  1 !
; 1121          2056  1 ! Implicit Outputs:
; 1122          2057  1 !               none
; 1123          2058  1 !
; 1124          2059  1 ! Returned Value:
; 1125          2060  1 !               none
; 1126          2061  1 !
; 1127          2062  1 ! Side Effects:
; 1128          2063  1 !               none
; 1129          2064  1 !--
; 1130          2065  1 ROUTINE DEALLOCATE_PAGE (
; 1131          2066  1                     SCB : REF $BBLOCK
; 1132          2067  1                    ) =
; 1133          2068  2 BEGIN
; 1134          2069  2     LOCAL
; 1135          2070  2         PAGE_SIZE;
; 1136          2071  2
; 1137          2072  2 PAGE_SIZE = .SCB[PSM$L_PAGE_WIDTH] * (.SCB[PSM$L_PAGE_LENGTH]+1);
; 1138          2073  2
; 1139        P 2074  2 RETURN_IF_ERROR_( LIB$FREE_VM ( %REF(.PAGE_SIZE),
; 1140          2075  2                       SCB[PSM$A_PAGE_POINTER]));
; 1141          2076  2
; 1142          2077  2 RETURN SS$_NORMAL;
; 1143          2078  1 END;



                                                       0000 00000 DEALLOCATE_PAGE:
                                                                           .WORD   Save nothing                                 ; 2065
                                       5E             04  C2 00002         SUBL2   #4, SP                                       :
                                       51       04    AC  D0 00005         MOVL    SCB, R1                                      ; 2072
                         50     01F8   C1             01  C1 00009         ADDL3   #1, 504(R1), R0                              :
                                       50     0200    C1  C4 0000F         MULL2   512(R1), PAGE_SIZE                           :
                                              01FC    C1  9F 00014         PUSHAB  508(R1)                                      ; 2075
                                  04   AE             50  D0 00018         MOVL    PAGE_SIZE, 4(SP)                             :
                                                04    AE  9F 0001C         PUSHAB  4(SP)                                        :
                            00000000G  00             02  FB 0001F         CALLS   #2, LIB$FREE_VM                              :
                                       03             50  E9 00026         BLBC    STATUS, 1$                                   :
                                       50             01  D0 00029         MOVL    #1, R0                                       ; 2077
                                                         04 0002C 1$:      RET                                                  ; 2078

; Routine Size:  45 bytes,    Routine Base:  CODE + 0699
```

```
%sbttl 'CREATE_PAGE_HEADER - Allocate and Format the Page Header'
!++
! Functional Description:
!       This routine allocates memory and formats the information
!       for the page header.  Returns success if allocation of memory
!       was successful.
!
! Formal Parameters:
!               SCB     - Address of the SCB
!
! Implicit Inputs:
!               none
!
! Implicit Outputs:
!               none
!
! Returned Value:
!               none
!
! Side Effects:
!               none
!--
ROUTINE CREATE_PAGE_HEADER (
                    SCB : REF $BBLOCK
                   ) =
BEGIN

LOCAL
    REMAINING,                                          ! Remaining header space
    NAME_LENGTH ,                                       ! Trimmed file name length
    BUFFER      : VECTOR [512,byte],                    ! Assume max size 512 bytes0
    STR_DESC    : VECTOR [2],
    HEADER_REF  : REF PAGE_ARRAY,
    HEADER_SIZE ;

!*! SMALL WIDTHS -- THE PAGE NUMBER SHOULD BE THE ONLY THING PRINTED
!*! WHEN THE WIDTH IS TOO SMALL.  DATE vs. FILENAME IS DEVO'S CHOICE

HEADER_SIZE = .SCB_SIZE_ (PAGE_HEADER) - 3;             ! don't include the carriage
                                                        !    control area of 3 bytes
HEADER_REF  = .SCB_ADDR_ (PAGE_HEADER);

CH$FILL (%CHAR(32), .HEADER_SIZE, .HEADER_REF);

! Insert imbedded carriage control <LF><LF><CR>
!
CH$FILL (PSM$K_CHAR_LF, 2,(.HEADER_REF + .HEADER_SIZE));
CH$FILL (PSM$K_CHAR_CR, 1,(.HEADER_REF + .HEADER_SIZE) + 2);
                                                        ! address is offset by two

! If the header is too small even for "Page 99999" then disable page
! headers.  (Maybe this code should be in "MESSAGE").
!
IF .HEADER_SIZE LSSU 10
THEN
    RETURN SS$_NORMAL;

```

```
! Set up the buffer descriptor for "GET_xxx" Routines
!
STR_DESC[SIZE] = %ALLOCATION(BUFFER);                   ! allocate for routines
STR_DESC[ADDR] = BUFFER;                                ! init address

! Insert the word "Page "
!
MOVE_FRAME
        (.SCB,
        $DESCRIPTOR ('Page '),
        HEADER_REF[.HEADER_SIZE-10,0,.SCB[PSM$L_FORM_WIDTH]],
        5,
        1);

! Get the filename - include the expected length
!
STR_DESC[SIZE] = %ALLOCATION(BUFFER);                   ! reset buffer length
GET_FILE_NAME
        (.SCB,                                          ! SCB addr.
        .HEADER_SIZE - 11,                              ! expected length (less page #)
        STR_DESC[0],                                    ! Buffer descriptor
        STR_DESC[SIZE]);                                ! Returned length

NAME_LENGTH = .STR_DESC[SIZE] + 1;                      ! Save the trimmed length

INSERT_FRAME                                            ! Left Justified
        (.SCB,
        STR_DESC[0],
        HEADER_REF[0,0,.SCB[PSM$L_FORM_WIDTH]],
        .NAME_LENGTH - 1,                               ! Always less than frame_width
        1);


! The area remaining for the date is the original header
! width less the size of the file name, less the size for the page
! number field ("Page 99999") less one blank for each.
!
REMAINING = .HEADER_SIZE - .NAME_LENGTH - 10 - 1;
IF .REMAINING LSSU 18
THEN
    RETURN SS$_NORMAL;

! Get the file revision date and center it between file name and page number
!
STR_DESC[SIZE] = %ALLOCATION(BUFFER);                   ! reset buffer length
GET_REVISION_DATE
        (.SCB,                                          ! SCB addr.
        STR_DESC[0],                                    ! Buffer descriptor
        STR_DESC[SIZE]);                                ! Returned length

CENTER_FRAME
        (.SCB,
        STR_DESC[0],
        HEADER_REF[.NAME_LENGTH,0,.SCB[PSM$L_FORM_WIDTH]],
        .REMAINING,
        1);

```

```
; 1259          2193  2 RETURN SS$_NORMAL;
; 1260          2194  2
; 1261          2195  1 END;



                                         20  65   67   61   50 006C6 P.AAD:  .ASCII  \Page \                                        ;
                                                               006CB         .BLKB   1
                                                    00000005   006CC P.AAC:  .LONG   5                                              ;
                                                    00000000'  006D0         .ADDRESS P.AAD                                         ;



                                                          01FC 00000 CREATE_PAGE_HEADER:
                                                                             .WORD   Save R2,R3,R4,R5,R6,R7,R8                      ; 2101
                                          5E     FDF8   CE  9E 00002         MOVAB   -520(SP), SP                                   ;
                                          58       04   AC  D0 00007         MOVL    SCB, R8                                        ; 2117
                                          50     01F0   C8  9E 0000B         MOVAB   496(R8), R0                                    ;
                                          56            60  3C 00010         MOVZWL  (R0), HEADER_SIZE                              ;
                                          56            03  C2 00013         SUBL2   #3, HEADER_SIZE                                ;
                                          57       04   A0  D0 00016         MOVL    4(R0), HEADER_REF                              ; 2119
           56             20              6E            00  2C 0001A         MOVC5   #0, (SP), #32, HEADER_SIZE, (HEADER_REF)       ; 2121
                                                        67     0001F
                                                      6647  9F 00020         PUSHAB  (HEADER_SIZE)[HEADER_REF]                      ; 2125
                                          9E     0A0A   8F  B0 00023         MOVW    #2570, @(SP)+                                  ;
                          50              57            56  C1 00028         ADDL3   HEADER_SIZE, HEADER_REF, R0                    ;
                                   02     A0            0D  90 0002C         MOVB    #13, 2(R0)                                     ; 2126
                                          0A            56  D1 00030         CMPL    HEADER_SIZE, #10                               ; 2132
                                                        72  1F 00033         BLSSU   1$                                             ;
                                          6E     0200   8F  3C 00035         MOVZWL  #512, STR_DESC                                 ; 2138
                                   04     AE       08   AE  9E 0003A         MOVAB   BUFFER, STR_DESC+4                             ; 2139
                                                        01  DD 0003F         PUSHL   #1                                             ; 2146
                                                        05  DD 00041         PUSHL   #5                                             ;
                                                  F6  A647  9F 00043         PUSHAB  -10(HEADER_SIZE)[HEADER_REF]                   ;
                                                  AE    AF  9F 00047         PUSHAB  P.AAC                                          ; 2145
                                                        58  DD 0004A         PUSHL   R8                                             ; 2146
                                0000V     CF            05  FB 0004C         CALLS   #5, MOVE_FRAME                                 ;
                                          6E     0200   8F  3C 00051         MOVZWL  #512, STR_DESC                                 ; 2152
                                                        5E  DD 00056         PUSHL   SP                                             ; 2157
                                                  04    AE  9F 00058         PUSHAB  STR_DESC                                       ; 2156
                                                  F5    A6  9F 0005B         PUSHAB  -11(HEADER_SIZE)                               ; 2155
                                                        58  DD 0005E         PUSHL   R8                                             ; 2154
                                0000V     CF            04  FB 00060         CALLS   #4, GET_FILE_NAME                              ;
                          52              6E            01  C1 00065         ADDL3   #1, STR_DESC, NAME_LENGTH                      ; 2159
                                                        01  DD 00069         PUSHL   #1                                             ; 2164
                                                  FF    A2  9F 0006B         PUSHAB  -1(NAME_LENGTH)                                ; 2165
                                                        57  DD 0006E         PUSHL   HEADER_REF                                     ; 2164
                                                  0C    AE  9F 00070         PUSHAB  STR_DESC                                       ; 2163
                                                        58  DD 00073         PUSHL   R8                                             ; 2164
                                0000V     CF            05  FB 00075         CALLS   #5, INSERT_FRAME                               ;
                                          56            52  C2 0007A         SUBL2   NAME_LENGTH, R6                                ; 2173
                                          56            0B  C2 0007D         SUBL2   #11, REMAINING                                 ;
                                          12            56  D1 00080         CMPL    REMAINING, #18                                 ; 2174
                                                        22  1F 00083         BLSSU   1$                                             ;
                                          6E     0200   8F  3C 00085         MOVZWL  #512, STR_DESC                                 ; 2180
                                                        5E  DD 0008A         PUSHL   SP                                             ; 2184
                                                  04    AE  9F 0008C         PUSHAB  STR_DESC                                       ; 2183
                                          58  DD 0008F          PUSHL   R8                                          ; 2182
                        0000V  CF         03  FB 00091          CALLS   #3, GET_REVISION_DATE                       ;
                                          01  DD 00096          PUSHL   #1                                          ; 2189
                                          56  DD 00098          PUSHL   REMAINING                                   ; 2190
                                        6247  9F 0009A          PUSHAB  (NAME_LENGTH)[HEADER_REF]                   ; 2189
                                    0C    AE  9F 0009D          PUSHAB  STR_DESC                                    ; 2188
                                          58  DD 000A0          PUSHL   R8                                          ; 2189
                        0000V  CF         05  FB 000A2          CALLS   #5, CENTER_FRAME                            ;
                               50         01  D0 000A7  1$:     MOVL    #1, R0                                      ; 2193
                                              04 000AA          RET                                                 ; 2195

; Routine Size:  171 bytes,    Routine Base:  CODE + 06D4
```

```
1 %sbttl 'FILL_FILE_FLAG - Insert Information into the FILE Page'
1 !++
1 ! Functional Description:
1 !         This procedure controls all inserts required for the FILE Page.
1 !
1 ! Formal Parameters:
1 !         SCB             - Address of the SCB
1 !         PAGE_REF        - Pointer to the Page (first byte)
1 !         PAGE_LENGTH     - Length of Frame
1 !         PAGE_WIDTH      - Width of Frame
1 !
1 ! Implicit Inputs:
1 !                 none
1 !
1 ! Implicit Outputs:
1 !                 none
1 !
1 ! Returned Value:
1 !                 none
1 !
1 ! Side Effects:
1 !                 none
1 !--
1 ROUTINE FILL_FILE_FLAG (
1         SCB             : REF $BBLOCK,
1         PAGE_REF        : REF PAGE_ARRAY,
1         PAGE_WIDTH,
1         PAGE_LENGTH
1         ):      NOVALUE =
2 BEGIN
2
2 LITERAL K_MAX_BUFFER_SIZE = 512;
2
2 LOCAL
2     RET_LEN     : VECTOR[1],
2     TOP_OFFSET  ,
2     BOTTOM_OFFSET,
2     BUFFER      : VECTOR [512,byte],              ! Assume max size 512 bytes
2     STRING_DESC : VECTOR [2];                     ! Descriptor to current string
2
2 ! Allocate the buffer for "GET_xxx" Routines
2 !
2 STRING_DESC[SIZE] = %ALLOCATION(BUFFER);          ! allocate for routines
2 STRING_DESC[ADDR] = BUFFER;                       ! init address
2
2 TOP_OFFSET = 0;                                   ! start insert at zero
2 BOTTOM_OFFSET = .PAGE_LENGTH - 2;                 ! Note: offset includes next
2                                                   !    "insert" frame length
2
2 ! Burst characters
2 !
2     FILL_FRAME (.SCB,
2                 .SCB[PSM$B_FILE_BURST_CHAR],
2                 PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH, 3);
2
2     FILL_FRAME (.SCB,
2                 .SCB[PSM$B_FILE_BURST_CHAR],
                PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH, 3);

    FILL_FRAME (.SCB,
                %C' ',
                PAGE_REF[10,.TOP_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH-20, 3);

    FILL_FRAME (.SCB,
                %C' ',
                PAGE_REF[10,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH-20, 3);

    FILL_FRAME (.SCB,
                .SCB[PSM$B_JOB_BURST_CHAR],
                PAGE_REF[14,.TOP_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH-28, 3);

    FILL_FRAME (.SCB,
                .SCB[PSM$B_JOB_BURST_CHAR],
                PAGE_REF[14,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH-28, 3);

! Get the sys$announce note and output to page
!       note:   system announcement will fit or will be truncated so there is
!               no updating of "offsets"
!
    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE; ! reset buffer size

    GET_SYSTEM_ANNOUNCEMENT
                (.SCB,                  ! SCB addr.
                STRING_DESC[0],         ! Buffer descriptor
                STRING_DESC[SIZE]);     ! Returned length

    CENTER_FRAME (.SCB,
                   STRING_DESC[0],
                   PAGE_REF[0,.TOP_OFFSET + 1,.PAGE_WIDTH], .PAGE_WIDTH, 1);

    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE; ! reset buffer size

    GET_DIGITAL_LOGO
                (.SCB,                  ! SCB addr.
                STRING_DESC[0],         ! Buffer descriptor
                STRING_DESC[SIZE]);     ! Returned length

    CENTER_FRAME (.SCB,
                   STRING_DESC[0],
                   PAGE_REF[0,.BOTTOM_OFFSET + 1,.PAGE_WIDTH], .PAGE_WIDTH, 1);


! Create a sentence describing the current job.
!
    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size
    TOP_OFFSET = .TOP_OFFSET + 4;               ! adjust & allow for spacing

    GET_JOB_DESCRIPTION
                (.SCB,                  ! SCB addr.
                 1,                     ! Use present tense
                 STRING_DESC[0],        ! Buffer descriptor
                 STRING_DESC[SIZE]);     ! Returned length

    RET_LEN[0] = RETURN_FRAME_LENGTH
                 (.SCB,
                  STRING_DESC[0],                ! string ref.
                  PAGE_REF[0,0,.PAGE_WIDTH],     ! ref to frame
                  .PAGE_WIDTH,                   ! cols to fill
                  .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill

    IF .RET_LEN[0] GTR 0
    THEN
        BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
                                                 ! offset before inserting
                                                 !   includes the space
    ! Insert the string delimited.  Bottom of page.
    INSERT_FRAME (.SCB,
                  STRING_DESC[0],                ! string ref.
                  PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH],
                                                 ! ref to frame
                  .PAGE_WIDTH,                   ! cols to fill
                  .RET_LEN[0]);                  ! rows to fill

!  Create a sentence describing the current file.  Bottom of page.
!
    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size

    GET_FILE_DESCRIPTION
                (.SCB,                           ! SCB addr.
                 STRING_DESC[0],                 ! Buffer descriptor
                 STRING_DESC[SIZE]);             ! Returned length

    RET_LEN[0] = RETURN_FRAME_LENGTH
                 (.SCB,
                  STRING_DESC[0],                ! string ref.
                  PAGE_REF[0,0,.PAGE_WIDTH],     ! ref to frame
                  .PAGE_WIDTH,                   ! cols to fill
                  .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill

    IF .RET_LEN[0] GTR 0
    THEN
        BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
                                                 ! offset before inserting

    ! insert the string delimited
    INSERT_FRAME (.SCB,
                  STRING_DESC[0],                ! string ref.
                  PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH],
                                                 ! ref to frame
                  .PAGE_WIDTH,                   ! cols to fill
                  .RET_LEN[0]);                  ! rows to fill

! User note
!
    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
```

```
; 1434          2367  2       !  Get the user note
; 1435          2368  2       GET_USER_NOTE
; 1436          2369  2                   (.SCB,                          ! SCB addr.
; 1437          2370  2                   STRING_DESC[0],                 ! Buffer descriptor
; 1438          2371  2                   STRING_DESC[SIZE]);             ! Returned length
; 1439          2372  2
; 1440          2373  2       RET_LEN[0] = RETURN_FRAME_LENGTH
; 1441          2374  2                   (.SCB,
; 1442          2375  2                    STRING_DESC[0],                ! string ref.
; 1443          2376  2                    PAGE_REF[0,0,.PAGE_WIDTH],     ! ref to frame
; 1444          2377  2                    .PAGE_WIDTH,                   ! cols to fill
; 1445          2378  2                    .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill
; 1446          2379  2
; 1447          2380  2       ! insert the string delimited
; 1448          2381  2       INSERT_FRAME (.SCB,
; 1449          2382  2                    STRING_DESC[0],                ! string ref.
; 1450          2383  2                    PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 1451          2384  2                                                   ! ref to frame
; 1452          2385  2                    .PAGE_WIDTH,                   ! cols to fill
; 1453          2386  2                    .RET_LEN[0]);                  ! rows to fill
; 1454          2387  2
; 1455          2388  2 ! User name
; 1456          2389  2 !
; 1457          2390  2       ! re-init
; 1458          2391  2       STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
; 1459          2392  2       IF .RET_LEN[0] GTR 0
; 1460          2393  2       THEN
; 1461          2394  2           TOP_OFFSET      = .TOP_OFFSET + .RET_LEN[0] + 1
; 1462          2395  2       ELSE
; 1463          2396  2           TOP_OFFSET      = .TOP_OFFSET + 1;
; 1464          2397  2                                                   ! adjust & allow for spacing
; 1465          2398  2
; 1466          2399  2       RET_LEN[0] = INSERT_NAME_BANNER (.SCB,
; 1467          2400  2                           SCB_SIZE (USER_NAME),   ! user name desc
; 1468          2401  2                           PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 1469          2402  2                                                   ! ref to frame
; 1470          2403  2                           .PAGE_WIDTH,            ! max width Bann
; 1471          2404  2                           .BOTTOM_OFFSET - .TOP_OFFSET,
; 1472          2405  2                                                   ! frame length
; 1473          2406  2                           7);                     ! max hght Bann str desired
; 1474          2407  2
; 1475          2408  2       IF .RET_LEN[0] GTR 0
; 1476          2409  2       THEN
; 1477          2410  2           TOP_OFFSET      = .TOP_OFFSET + .RET_LEN[0] + 2;
; 1478          2411  2                                                   ! adjust & allow for spacing
; 1479          2412  2
; 1480          2413  2 !  Get and insert the filename banner
; 1481          2414  2 !
; 1482          2415  2       ! re-init
; 1483          2416  2       STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
; 1484          2417  2       RET_LEN[0] = INSERT_FILENAME_BANNER
; 1485          2418  2                           (.SCB,
; 1486          2419  2                            STRING_DESC[0],        ! Buffer desc.
; 1487          2420  2                            PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 1488          2421  2                                                   ! ref to frame
; 1489          2422  2                            .PAGE_WIDTH,           ! max width Bann
; 1490          2423  2                             .BOTTOM_OFFSET - .TOP_OFFSET);
; 1491          2424  2                                         ! rows to fill
; 1492          2425  2         IF .RET_LEN[0] GTR 0
; 1493          2426  2         THEN
; 1494          2427  2             TOP_OFFSET      = .TOP_OFFSET + .RET_LEN[0] + 2;
; 1495          2428  2                                         ! adjust & allow for spacing
; 1496          2429  2
; 1497          2430  1 END;



                                        OFFC 00000 FILL_FILE_FLAG:
                                                        .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11                     ; 2219
                   5B     0000V   CF  9E 00002          MOVAB   INSERT_FRAME, R11                                        ;
                   5A     0000V   CF  9E 00007          MOVAB   RETURN_FRAME_LENGTH, R10                                 ;
                   59     0000V   CF  9E 0000C          MOVAB   FILL_FRAME, R9                                           ;
                   5E      FDFC   CE  9E 00011          MOVAB   -516(SP), SP                                             ;
                   7E      0200   8F  3C 00016          MOVZWL  #512, STRING_DESC                                        ; 2238
          04       AE        08   AE  9E 0001B          MOVAB   BUFFER, STRING_DESC+4                                    ; 2239
                                  52  D4 00020          CLRL    TOP_OFFSET                                               ; 2241
     53   10       AC             02  C3 00022          SUBL3   #2, PAGE_LENGTH, BOTTOM_OFFSET                           ; 2242
                                  03  DD 00027          PUSHL   #3                                                       ; 2249
                   57        0C   AC  D0 00029          MOVL    PAGE_WIDTH, R7                                           ;
                                  57  DD 0002D          PUSHL   R7                                                       ;
                   54        08   AC  D0 0002F          MOVL    PAGE_REF, R4                                             ;
     58            52             57  C5 00033          MULL3   R7, TOP_OFFSET, R8                                       ;
                                6844  9F 00037          PUSHAB  (R8)[R4]                                                 ;
                   56        04   AC  D0 0003A          MOVL    SCB, R6                                                  ; 2248
                   7E      02A4   C6  9A 0003E          MOVZBL  676(R6), -(SP)                                           ; 2249
                                  56  DD 00043          PUSHL   R6                                                       ;
                   69             05  FB 00045          CALLS   #5, FILL_FRAME                                           ;
                                  03  DD 00048          PUSHL   #3                                                       ; 2253
                                  57  DD 0004A          PUSHL   R7                                                       ;
     55            53             57  C5 0004C          MULL3   R7, BOTTOM_OFFSET, R5                                    ;
                                6544  9F 00050          PUSHAB  (R5)[R4]                                                 ;
                   7E      02A4   C6  9A 00053          MOVZBL  676(R6), -(SP)                                           ;
                                  56  DD 00058          PUSHL   R6                                                       ;
                   69             05  FB 0005A          CALLS   #5, FILL_FRAME                                           ;
                                  03  DD 0005D          PUSHL   #3                                                       ; 2257
                             EC   A7  9F 0005F          PUSHAB  -20(R7)                                                  ;
                             0A A844  9F 00062          PUSHAB  10(R8)[R4]                                               ;
                                  20  DD 00066          PUSHL   #32                                                      ;
                                  56  DD 00068          PUSHL   R6                                                       ;
                   69             05  FB 0006A          CALLS   #5, FILL_FRAME                                           ;
                                  03  DD 0006D          PUSHL   #3                                                       ; 2261
                             EC   A7  9F 0006F          PUSHAB  -20(R7)                                                  ;
                             0A A544  9F 00072          PUSHAB  10(R5)[R4]                                               ;
                                  20  DD 00076          PUSHL   #32                                                      ;
                                  56  DD 00078          PUSHL   R6                                                       ;
                   69             05  FB 0007A          CALLS   #5, FILL_FRAME                                           ;
                                  03  DD 0007D          PUSHL   #3                                                       ; 2265
                             E4   A7  9F 0007F          PUSHAB  -28(R7)                                                  ;
                             0E A844  9F 00082          PUSHAB  14(R8)[R4]                                               ;
                   7E      02A6   C6  9A 00086          MOVZBL  678(R6), -(SP)                                           ;
                                  56  DD 0008B          PUSHL   R6                                                       ;
                   69             05  FB 0008D          CALLS   #5, FILL_FRAME                                           ;
```

```
                         03  DD 00090         PUSHL   #3                                      ; 2269
                    E4   A7  9F 00092         PUSHAB  -28(R7)                                 ;
                    0E A544  9F 00095         PUSHAB  14(R5)[R4]                              ;
             7E   02A6   C6  9A 00099         MOVZBL  678(R6), -(SP)                          ;
                         56  DD 0009E         PUSHL   R6                                      ;
             69          05  FB 000A0         CALLS   #5, FILL_FRAME                          ;
             6E   0200   8F  3C 000A3         MOVZWL  #512, STRING_DESC                       ; 2276
                         5E  DD 000A8         PUSHL   SP                                      ; 2281
                    04   AE  9F 000AA         PUSHAB  STRING_DESC                             ; 2280
                         56  DD 000AD         PUSHL   R6                                      ; 2279
      0000V  CF          03  FB 000AF         CALLS   #3, GET_SYSTEM_ANNOUNCEMENT             ;
                         01  DD 000B4         PUSHL   #1                                      ; 2285
                         57  DD 000B6         PUSHL   R7                                      ;
             58     01   A2  9E 000B8         MOVAB   1(R2), R8                               ;
             58          57  C4 000BC         MULL2   R7, R8                                  ;
                       6844  9F 000BF         PUSHAB  (R8)[R4]                                ;
                    0C   AE  9F 000C2         PUSHAB  STRING_DESC                             ; 2284
                         56  DD 000C5         PUSHL   R6                                      ; 2285
      0000V  CF          05  FB 000C7         CALLS   #5, CENTER_FRAME                        ;
             6E   0200   8F  3C 000CC         MOVZWL  #512, STRING_DESC                       ; 2288
                         5E  DD 000D1         PUSHL   SP                                      ; 2293
                    04   AE  9F 000D3         PUSHAB  STRING_DESC                             ; 2292
                         56  DD 000D6         PUSHL   R6                                      ; 2291
      0000V  CF          03  FB 000D8         CALLS   #3, GET_DIGITAL_LOGO                    ;
                         01  DD 000DD         PUSHL   #1                                      ; 2297
                         57  DD 000DF         PUSHL   R7                                      ;
             55     01   A3  9E 000E1         MOVAB   1(R3), R5                               ;
             55          57  C4 000E5         MULL2   R7, R5                                  ;
                       6544  9F 000E8         PUSHAB  (R5)[R4]                                ;
                    0C   AE  9F 000EB         PUSHAB  STRING_DESC                             ; 2296
                         56  DD 000EE         PUSHL   R6                                      ; 2297
      0000V  CF          05  FB 000F0         CALLS   #5, CENTER_FRAME                        ;
             6E   0200   8F  3C 000F5         MOVZWL  #512, STRING_DESC                       ; 2303
             52          04  C0 000FA         ADDL2   #4, TOP_OFFSET                          ; 2304
                         5E  DD 000FD         PUSHL   SP                                      ; 2310
                    04   AE  9F 000FF         PUSHAB  STRING_DESC                             ; 2309
                         01  DD 00102         PUSHL   #1                                      ; 2307
                         56  DD 00104         PUSHL   R6                                      ;
      0000V  CF          04  FB 00106         CALLS   #4, GET_JOB_DESCRIPTION                 ;
7E           53          52  C3 0010B         SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)        ; 2317
                  0090   8F  BB 0010F         PUSHR   #^M<R4,R7>                              ; 2315
                    0C   AE  9F 00113         PUSHAB  STRING_DESC                             ; 2314
                         56  DD 00116         PUSHL   R6                                      ; 2315
             6A          05  FB 00118         CALLS   #5, RETURN_FRAME_LENGTH                 ;
             55          50  D0 0011B         MOVL    R0, RET_LEN                             ;
                         08  15 0011E         BLEQ    1$                                      ; 2319
50           53          55  C3 00120         SUBL3   RET_LEN, BOTTOM_OFFSET, R0              ; 2321
             53     FF   A0  9E 00124         MOVAB   -1(R0), BOTTOM_OFFSET                   ;
                         55  DD 00128  1$:    PUSHL   RET_LEN                                 ; 2330
                         57  DD 0012A         PUSHL   R7                                      ; 2329
50           53          57  C5 0012C         MULL3   R7, BOTTOM_OFFSET, R0                   ; 2327
                       6044  9F 00130         PUSHAB  (R0)[R4]                                ;
                    0C   AE  9F 00133         PUSHAB  STRING_DESC                             ; 2326
                         56  DD 00136         PUSHL   R6                                      ; 2327
             6B          05  FB 00138         CALLS   #5, INSERT_FRAME                        ;
             6E   0200   8F  3C 0013B         MOVZWL  #512, STRING_DESC                       ; 2335
                         5E  DD 00140         PUSHL   SP                                      ; 2340
```

```
                             04  AE  9F 00142          PUSHAB  STRING_DESC                              ; 2339
                                 56  DD 00145          PUSHL   R6                                       ; 2338
              0000V   CF         03  FB 00147          CALLS   #3, GET_FILE_DESCRIPTION                 ;
        7E            53         52  C3 0014C          SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)         ; 2347
                           0090  8F  BB 00150          PUSHR   #^M<R4,R7>                               ; 2345
                             0C  AE  9F 00154          PUSHAB  STRING_DESC                              ; 2344
                                 56  DD 00157          PUSHL   R6                                       ; 2345
                      6A         05  FB 00159          CALLS   #5, RETURN_FRAME_LENGTH                  ;
                      55         50  D0 0015C          MOVL    R0, RET_LEN                              ;
                                 08  15 0015F          BLEQ    2$                                       ; 2349
        50            53         55  C3 00161          SUBL3   RET_LEN, BOTTOM_OFFSET, R0               ; 2351
                      53     FF  A0  9E 00165          MOVAB   -1(R0), BOTTOM_OFFSET                    ;
                                 55  DD 00169 2$:      PUSHL   RET_LEN                                  ; 2360
                                 57  DD 0016B          PUSHL   R7                                       ; 2359
        50            53         57  C5 0016D          MULL3   R7, BOTTOM_OFFSET, R0                    ; 2357
                               6044  9F 00171          PUSHAB  (R0)[R4]                                 ;
                             0C  AE  9F 00174          PUSHAB  STRING_DESC                              ; 2356
                                 56  DD 00177          PUSHL   R6                                       ; 2357
                      6B         05  FB 00179          CALLS   #5, INSERT_FRAME                         ;
                      6E   0200  8F  3C 0017C          MOVZWL  #512, STRING_DESC                        ; 2365
                                 5E  DD 00181          PUSHL   SP                                       ; 2371
                             04  AE  9F 00183          PUSHAB  STRING_DESC                              ; 2370
                                 56  DD 00186          PUSHL   R6                                       ; 2369
              0000V   CF         03  FB 00188          CALLS   #3, GET_USER_NOTE                        ;
        7E            53         52  C3 0018D          SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)         ; 2378
                           0090  8F  BB 00191          PUSHR   #^M<R4,R7>                               ; 2376
                             0C  AE  9F 00195          PUSHAB  STRING_DESC                              ; 2375
                                 56  DD 00198          PUSHL   R6                                       ; 2376
                      6A         05  FB 0019A          CALLS   #5, RETURN_FRAME_LENGTH                  ;
                      55         50  D0 0019D          MOVL    R0, RET_LEN                              ;
                                 55  DD 001A0          PUSHL   RET_LEN                                  ; 2386
                                 57  DD 001A2          PUSHL   R7                                       ; 2385
        58            52         57  C5 001A4          MULL3   R7, TOP_OFFSET, R8                       ; 2383
                               6844  9F 001A8          PUSHAB  (R8)[R4]                                 ;
                             0C  AE  9F 001AB          PUSHAB  STRING_DESC                              ; 2382
                                 56  DD 001AE          PUSHL   R6                                       ; 2383
                      6B         05  FB 001B0          CALLS   #5, INSERT_FRAME                         ;
                      6E   0200  8F  3C 001B3          MOVZWL  #512, STRING_DESC                        ; 2391
                                 55  D5 001B8          TSTL    RET_LEN                                  ; 2392
                                 07  15 001BA          BLEQ    3$                                       ;
                      52     01 A542  9E 001BC         MOVAB   1(RET_LEN)[TOP_OFFSET], TOP_OFFSET       ; 2394
                                 02  11 001C1          BRB     4$                                       ;
                                 52  D6 001C3 3$:      INCL    TOP_OFFSET                               ; 2396
                                 07  DD 001C5 4$:      PUSHL   #7                                       ; 2401
        7E            53         52  C3 001C7          SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)         ; 2404
                                 57  DD 001CB          PUSHL   R7                                       ; 2403
        58            52         57  C5 001CD          MULL3   R7, TOP_OFFSET, R8                       ; 2401
                               6844  9F 001D1          PUSHAB  (R8)[R4]                                 ;
                           016C  C6  9F 001D4          PUSHAB  364(R6)                                  ; 2400
                                 56  DD 001D8          PUSHL   R6                                       ; 2401
              0000V   CF         06  FB 001DA          CALLS   #6, INSERT_NAME_BANNER                   ;
                      55         50  D0 001DF          MOVL    R0, RET_LEN                              ;
                                 05  15 001E2          BLEQ    5$                                       ; 2408
                      52     02 A542  9E 001E4         MOVAB   2(RET_LEN)[TOP_OFFSET], TOP_OFFSET       ; 2410
                      6E   0200  8F  3C 001E9 5$:      MOVZWL  #512, STRING_DESC                        ; 2416
        7E            53         52  C3 001EE          SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)         ; 2423
                                 57  DD 001F2          PUSHL   R7                                       ; 2422
```

```
                  58         52          57  C5 001F4          MULL3   R7, TOP_OFFSET, R8                     ; 2420
                                       6844  9F 001F8          PUSHAB  (R8)[R4]                               ;
                                  0C     AE  9F 001FB          PUSHAB  STRING_DESC                            ; 2419
                                         56  DD 001FE          PUSHL   R6                                     ; 2420
                      0000V  CF          05  FB 00200          CALLS   #5, INSERT_FILENAME_BANNER             ;
                             55          50  D0 00205          MOVL    R0, RET_LEN                            ;
                                         05  15 00208          BLEQ    6$                                     ; 2425
                             52     02 A542  9E 0020A          MOVAB   2(RET_LEN)[TOP_OFFSET], TOP_OFFSET     ; 2427
                                             04 0020F 6$:      RET                                            ; 2430

; Routine Size:  528 bytes,     Routine Base:  CODE + 077F
```

```
1 %sbttl 'FILL_JOB_FLAG - Insert Information into the JOB Page'
1 !++
1 ! Functional Description:
1 !       This procedure controls all inserts required for the JOB Page.
1 !
1 ! Formal Parameters:
1 !       SCB             - Address of the SCB
1 !       PAGE_REF        - Pointer to the Page (first byte)
1 !       PAGE_LENGTH     - Length of Frame
1 !       PAGE_WIDTH      - Width of Frame
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1 ROUTINE FILL_JOB_FLAG (
1       SCB             : REF $BBLOCK,
1       PAGE_REF        : REF PAGE_ARRAY,
1       PAGE_WIDTH,
1       PAGE_LENGTH
1       ):      NOVALUE =
2 BEGIN
2
2 LITERAL K_MAX_BUFFER_SIZE = 512;
2
2 LOCAL
2     RET_LEN     : VECTOR[1],
2     TOP_OFFSET  ,
2     BOTTOM_OFFSET,
2     BUFFER      : VECTOR [512,byte],                  ! Assume max size 512 bytes
2     STRING_DESC : VECTOR [2];                         ! Descriptor to current string
2
2 ! Allocate the buffer for "GET_xxx" Routines
2 !
2 STRING_DESC[SIZE] = %ALLOCATION(BUFFER);              ! allocate for routines
2 STRING_DESC[ADDR] = BUFFER;                           ! init address
2
2 TOP_OFFSET = 0;
2 BOTTOM_OFFSET = .PAGE_LENGTH - 2;                     ! offset includes burst offset
2
2 ! Burst Character
2 !
2     FILL_FRAME (.SCB,
2               .SCB[PSM$B_JOB_BURST_CHAR],
2               PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH, 3);
2
2     FILL_FRAME (.SCB,
2               .SCB[PSM$B_JOB_BURST_CHAR],
2               PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH, 3);
```

```
2
2 !  System announcement
2 !
2      ! re-init
2      STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
2      GET_SYSTEM_ANNOUNCEMENT
2                  (.SCB,                           ! SCB addr.
2                  STRING_DESC[0],                  ! Buffer descriptor
2                  STRING_DESC[SIZE]);              ! Returned length
2
2      CENTER_FRAME (.SCB,
2                     STRING_DESC[0],
2                     PAGE_REF[0,.TOP_OFFSET + 1,.PAGE_WIDTH], .PAGE_WIDTH, 1);
2
2      ! re-init
2      STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
2      GET_DIGITAL_LOGO
2                  (.SCB,                           ! SCB addr.
2                  STRING_DESC[0],                  ! Buffer descriptor
2                  STRING_DESC[SIZE]);              ! Returned length
2
2      CENTER_FRAME (.SCB,
2                     STRING_DESC[0],
2                     PAGE_REF[0,.BOTTOM_OFFSET + 1,.PAGE_WIDTH], .PAGE_WIDTH, 1);
2
2      TOP_OFFSET = .TOP_OFFSET + 4;                ! adjust & allow for spacing
2
2 !  Job description - create a sentence describing the current job.
2 !
2      ! re-init
2      STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
2      GET_JOB_DESCRIPTION
2                  (.SCB,                           ! SCB addr.
2                   1,                              ! Use present tense
2                   STRING_DESC[0],                 ! Buffer descriptor
2                   STRING_DESC[SIZE]);             ! Returned length
2
2      RET_LEN[0] = RETURN_FRAME_LENGTH
2                  (.SCB,
2                   STRING_DESC[0],                 ! string ref.
2                   PAGE_REF[0,0,.PAGE_WIDTH],      ! ref to frame
2                   .PAGE_WIDTH,                    ! cols to fill
2                   .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill
2
2      IF .RET_LEN[0] GTR 0
2      THEN
2          BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
2                                                   ! offset before inserting
2      ! insert the string delimited
2      INSERT_FRAME (.SCB,
2                    STRING_DESC[0],                ! string ref.
2                    PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH],
2                                                   ! ref to frame
2                    .PAGE_WIDTH,                   ! cols to fill
2                    .RET_LEN[0]);                  ! rows to fill
2
2 ! User note
```

```
; 1613      2545  2 !
; 1614      2546  2     ! re-init
; 1615      2547  2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;          ! reset buffer size
; 1616      2548  2
; 1617      2549  2     !  Get the user note
; 1618      2550  2     GET_USER_NOTE
; 1619      2551  2                 (.SCB,                              ! SCB addr.
; 1620      2552  2                 STRING_DESC[0],                     ! Buffer descriptor
; 1621      2553  2                 STRING_DESC[SIZE]);                 ! Returned length
; 1622      2554  2
; 1623      2555  2     RET_LEN[0] = RETURN_FRAME_LENGTH
; 1624      2556  2                 (.SCB,
; 1625      2557  2                  STRING_DESC[0],                    ! string ref.
; 1626      2558  2                  PAGE_REF[0,0,.PAGE_WIDTH],         ! ref to frame
; 1627      2559  2                  .PAGE_WIDTH,                       ! cols to fill
; 1628      2560  2                  .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill
; 1629      2561  2
; 1630      2562  2     ! insert the string delimited
; 1631      2563  2     INSERT_FRAME (.SCB,
; 1632      2564  2                  STRING_DESC[0],                    ! string ref.
; 1633      2565  2                  PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 1634      2566  2                                                     ! ref to frame
; 1635      2567  2                  .PAGE_WIDTH,                       ! cols to fill
; 1636      2568  2                  .RET_LEN[0]);                      ! rows to fill
; 1637      2569  2
; 1638      2570  2     IF .RET_LEN[0] GTR 0
; 1639      2571  2     THEN
; 1640      2572  2         TOP_OFFSET       = .TOP_OFFSET + .RET_LEN[0] + 1
; 1641      2573  2     ELSE
; 1642      2574  2         TOP_OFFSET       = .TOP_OFFSET + 1;
; 1643      2575  2                                                     ! adjust & allow for spacing
; 1644      2576  2 ! User Name
; 1645      2577  2 !
; 1646      2578  2     ! re-init
; 1647      2579  2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;          ! reset buffer size
; 1648      2580  2
; 1649      2581  2     RET_LEN[0] = INSERT_NAME_BANNER (
; 1650      2582  2                         .SCB,
; 1651      2583  2                         SCB_SIZE (USER_NAME),       ! user name descriptor
; 1652      2584  2                         PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 1653      2585  2                                                     ! ref to frame
; 1654      2586  2                         .PAGE_WIDTH,                ! max width Bann
; 1655      2587  2                         .BOTTOM_OFFSET-.TOP_OFFSET,
; 1656      2588  2                                                     ! space left
; 1657      2589  2                         14);                        ! max hght Bann string desired
; 1658      2590  2
; 1659      2591  2     TOP_OFFSET  = .TOP_OFFSET + .RET_LEN[0] + 2;! adjust for banner & spacing
; 1660      2592  2
; 1661      2593  2 !  Job Name
; 1662      2594  2 !
; 1663      2595  2     ! re-init
; 1664      2596  2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;          ! reset buffer size
; 1665      2597  2
; 1666      2598  2     GET_JOB_NAME
; 1667      2599  2                 (.SCB,                              ! SCB addr.
; 1668      2600  2                 STRING_DESC[0],                     ! Buffer descriptor
; 1669      2601  2                 STRING_DESC[SIZE]);                 ! Returned length
; 1670      2602  2
; 1671      2603  2       RET_LEN[0] = INSERT_NAME_BANNER (
; 1672      2604  2                              .SCB,
; 1673      2605  2                              STRING_DESC[SIZE],          ! job name desc
; 1674      2606  2                              PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 1675      2607  2                                                          ! ref to frame
; 1676      2608  2                              .PAGE_WIDTH,                ! max width Bann
; 1677      2609  2                              .BOTTOM_OFFSET-.TOP_OFFSET,
; 1678      2610  2                                                          ! space left
; 1679      2611  2                              7);                         ! max hght Bann str desired
; 1680      2612  2
; 1681      2613  2       IF .RET_LEN[0] GTR 0
; 1682      2614  2       THEN
; 1683      2615  2           TOP_OFFSET        = .TOP_OFFSET + .RET_LEN[0] + 2;
; 1684      2616  2                                                          ! adjust & allow for spacing
; 1685      2617  2       !  Get and insert the filename banner
; 1686      2618  2       !
; 1687      2619  2
; 1688      2620  2       ! re-init
; 1689      2621  2       STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;          ! reset buffer size
; 1690      2622  2
; 1691      2623  2       IF (.BOTTOM_OFFSET - 9) GTR .TOP_OFFSET         ! test for enough room
; 1692      2624  2       THEN
; 1693      2625  3           BEGIN
; 1694      2626  3           BOTTOM_OFFSET = .BOTTOM_OFFSET - 9;
; 1695      2627  3                                                          ! offset before inserting
; 1696      2628  3           INSERT_JOBNUMBER_BANNER
; 1697      2629  3                   (.SCB,
; 1698      2630  3                    STRING_DESC[0],                        ! Buffer desc.
; 1699      2631  3                    PAGE_REF[0, .BOTTOM_OFFSET,.PAGE_WIDTH],
; 1700      2632  3                                                          ! ref to frame
; 1701      2633  3                    .PAGE_WIDTH,                           ! max width Bann
; 1702      2634  3                    7);                                    ! rows to fill
; 1703      2635  2           END;
; 1704      2636  1 END;



                                             00FC 00000 FILL_JOB_FLAG:
                                                        .WORD   Save R2,R3,R4,R5,R6,R7                    ; 2454
                   5E      FDFC    CE    9E 00002       MOVAB   -516(SP), SP                              ;
                   7E      0200    8F    3C 00007       MOVZWL  #512, STRING_DESC                         ; 2473
             04    AE        08    AE    9E 0000C       MOVAB   BUFFER, STRING_DESC+4                     ; 2474
                                   52    D4 00011       CLRL    TOP_OFFSET                                ; 2476
       53    10    AC              02    C3 00013       SUBL3   #2, PAGE_LENGTH, BOTTOM_OFFSET            ; 2477
                                   03    DD 00018       PUSHL   #3                                        ; 2483
                   57        0C    AC    D0 0001A       MOVL    PAGE_WIDTH, R7                            ;
                                   57    DD 0001E       PUSHL   R7                                        ;
                   54        08    AC    D0 00020       MOVL    PAGE_REF, R4                              ;
       50          52              57    C5 00024       MULL3   R7, TOP_OFFSET, R0                        ;
                           6044          9F 00028       PUSHAB  (R0)[R4]                                  ;
                   56        04    AC    D0 0002B       MOVL    SCB, R6                                   ; 2482
                   7E      02A6    C6    9A 0002F       MOVZBL  678(R6), -(SP)                            ; 2483
                                   56    DD 00034       PUSHL   R6                                        ;
          0000V    CF              05    FB 00036       CALLS   #5, FILL_FRAME                            ;
```

```
                                   03  DD 0003B          PUSHL   #3                                      ; 2487
                                   57  DD 0003D          PUSHL   R7                                      ;
      50              53           57  C5 0003F          MULL3   R7, BOTTOM_OFFSET, R0                   ;
                                 6044  9F 00043          PUSHAB  (R0)[R4]                                ;
                      7E    02A6   C6  9A 00046          MOVZBL  678(R6), -(SP)                          ;
                                   56  DD 0004B          PUSHL   R6                                      ;
              0000V   CF           05  FB 0004D          CALLS   #5, FILL_FRAME                          ;
                      6E    0200   8F  3C 00052          MOVZWL  #512, STRING_DESC                       ; 2492
                                   5E  DD 00057          PUSHL   SP                                      ; 2496
                              04   AE  9F 00059          PUSHAB  STRING_DESC                             ; 2495
                                   56  DD 0005C          PUSHL   R6                                      ; 2494
              0000V   CF           03  FB 0005E          CALLS   #3, GET_SYSTEM_ANNOUNCEMENT             ;
                                   01  DD 00063          PUSHL   #1                                      ; 2500
                                   57  DD 00065          PUSHL   R7                                      ;
                      50      01   A2  9E 00067          MOVAB   1(R2), R0                               ;
                      50           57  C4 0006B          MULL2   R7, R0                                  ;
                                 6044  9F 0006E          PUSHAB  (R0)[R4]                                ;
                              0C   AE  9F 00071          PUSHAB  STRING_DESC                             ; 2499
                                   56  DD 00074          PUSHL   R6                                      ; 2500
              0000V   CF           05  FB 00076          CALLS   #5, CENTER_FRAME                        ;
                      6E    0200   8F  3C 0007B          MOVZWL  #512, STRING_DESC                       ; 2503
                                   5E  DD 00080          PUSHL   SP                                      ; 2507
                              04   AE  9F 00082          PUSHAB  STRING_DESC                             ; 2506
                                   56  DD 00085          PUSHL   R6                                      ; 2505
              0000V   CF           03  FB 00087          CALLS   #3, GET_DIGITAL_LOGO                    ;
                                   01  DD 0008C          PUSHL   #1                                      ; 2511
                                   57  DD 0008E          PUSHL   R7                                      ;
                      50      01   A3  9E 00090          MOVAB   1(R3), R0                               ;
                      50           57  C4 00094          MULL2   R7, R0                                  ;
                                 6044  9F 00097          PUSHAB  (R0)[R4]                                ;
                              0C   AE  9F 0009A          PUSHAB  STRING_DESC                             ; 2510
                                   56  DD 0009D          PUSHL   R6                                      ; 2511
              0000V   CF           05  FB 0009F          CALLS   #5, CENTER_FRAME                        ;
                      52           04  C0 000A4          ADDL2   #4, TOP_OFFSET                          ; 2513
                      6E    0200   8F  3C 000A7          MOVZWL  #512, STRING_DESC                       ; 2518
                                   5E  DD 000AC          PUSHL   SP                                      ; 2523
                              04   AE  9F 000AE          PUSHAB  STRING_DESC                             ; 2522
                                   01  DD 000B1          PUSHL   #1                                      ; 2520
                                   56  DD 000B3          PUSHL   R6                                      ;
              0000V   CF           04  FB 000B5          CALLS   #4, GET_JOB_DESCRIPTION                 ;
      7E              53           52  C3 000BA          SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)        ; 2530
                            0090   8F  BB 000BE          PUSHR   #^M<R4,R7>                              ; 2528
                              0C   AE  9F 000C2          PUSHAB  STRING_DESC                             ; 2527
                                   56  DD 000C5          PUSHL   R6                                      ; 2528
              0000V   CF           05  FB 000C7          CALLS   #5, RETURN_FRAME_LENGTH                 ;
                      55           50  D0 000CC          MOVL    R0, RET_LEN                             ;
                                   08  15 000CF          BLEQ    1$                                      ; 2532
      50              53           55  C3 000D1          SUBL3   RET_LEN, BOTTOM_OFFSET, R0              ; 2534
                      53      FF   A0  9E 000D5          MOVAB   -1(R0), BOTTOM_OFFSET                   ;
                                   55  DD 000D9 1$:      PUSHL   RET_LEN                                 ; 2542
                                   57  DD 000DB          PUSHL   R7                                      ; 2541
      50              53           57  C5 000DD          MULL3   R7, BOTTOM_OFFSET, R0                   ; 2539
                                 6044  9F 000E1          PUSHAB  (R0)[R4]                                ;
                              0C   AE  9F 000E4          PUSHAB  STRING_DESC                             ; 2538
                                   56  DD 000E7          PUSHL   R6                                      ; 2539
              0000V   CF           05  FB 000E9          CALLS   #5, INSERT_FRAME                        ;
                      6E    0200   8F  3C 000EE          MOVZWL  #512, STRING_DESC                       ; 2547
```

```
                              5E DD 000F3        PUSHL    SP                                        ; 2553
                        04    AE 9F 000F5        PUSHAB   STRING_DESC                               ; 2552
                              56 DD 000F8        PUSHL    R6                                        ; 2551
      0000V   CF              03 FB 000FA        CALLS    #3, GET_USER_NOTE                         ;
7E            53              52 C3 000FF        SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)          ; 2560
                      0090    8F BB 00103        PUSHR    #^M<R4,R7>                                ; 2558
                        0C    AE 9F 00107        PUSHAB   STRING_DESC                               ; 2557
                              56 DD 0010A        PUSHL    R6                                        ; 2558
      0000V   CF              05 FB 0010C        CALLS    #5, RETURN_FRAME_LENGTH                   ;
              55              50 D0 00111        MOVL     R0, RET_LEN                               ;
                              55 DD 00114        PUSHL    RET_LEN                                   ; 2568
                              57 DD 00116        PUSHL    R7                                        ; 2567
50            52              57 C5 00118        MULL3    R7, TOP_OFFSET, R0                        ; 2565
                            6044 9F 0011C        PUSHAB   (R0)[R4]                                  ;
                        0C    AE 9F 0011F        PUSHAB   STRING_DESC                               ; 2564
                              56 DD 00122        PUSHL    R6                                        ; 2565
      0000V   CF              05 FB 00124        CALLS    #5, INSERT_FRAME                          ;
                              55 D5 00129        TSTL     RET_LEN                                   ; 2570
                              07 15 0012B        BLEQ     2$                                        ;
              52        01  A542 9E 0012D        MOVAB    1(RET_LEN)[TOP_OFFSET], TOP_OFFSET        ; 2572
                              02 11 00132        BRB      3$                                        ;
                              52 D6 00134 2$:    INCL     TOP_OFFSET                                ; 2574
              6E      0200    8F 3C 00136 3$:    MOVZWL   #512, STRING_DESC                         ; 2579
                              0E DD 0013B        PUSHL    #14                                       ; 2584
7E            53              52 C3 0013D        SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)          ; 2587
                              57 DD 00141        PUSHL    R7                                        ; 2586
50            52              57 C5 00143        MULL3    R7, TOP_OFFSET, R0                        ; 2584
                            6044 9F 00147        PUSHAB   (R0)[R4]                                  ;
                      016C    C6 9F 0014A        PUSHAB   364(R6)                                   ; 2583
                              56 DD 0014E        PUSHL    R6                                        ; 2584
      0000V   CF              06 FB 00150        CALLS    #6, INSERT_NAME_BANNER                    ;
              55              50 D0 00155        MOVL     R0, RET_LEN                               ;
              52        02  A542 9E 00158        MOVAB    2(RET_LEN)[TOP_OFFSET], TOP_OFFSET        ; 2591
              6E      0200    8F 3C 0015D        MOVZWL   #512, STRING_DESC                         ; 2596
                              5E DD 00162        PUSHL    SP                                        ; 2601
                        04    AE 9F 00164        PUSHAB   STRING_DESC                               ; 2600
                              56 DD 00167        PUSHL    R6                                        ; 2599
      0000V   CF              03 FB 00169        CALLS    #3, GET_JOB_NAME                          ;
                              07 DD 0016E        PUSHL    #7                                        ; 2606
7E            53              52 C3 00170        SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)          ; 2609
                              57 DD 00174        PUSHL    R7                                        ; 2608
50            52              57 C5 00176        MULL3    R7, TOP_OFFSET, R0                        ; 2606
                            6044 9F 0017A        PUSHAB   (R0)[R4]                                  ;
                        10    AE 9F 0017D        PUSHAB   STRING_DESC                               ; 2605
                              56 DD 00180        PUSHL    R6                                        ; 2606
      0000V   CF              06 FB 00182        CALLS    #6, INSERT_NAME_BANNER                    ;
              55              50 D0 00187        MOVL     R0, RET_LEN                               ;
                              05 15 0018A        BLEQ     4$                                        ; 2613
              52        02  A542 9E 0018C        MOVAB    2(RET_LEN)[TOP_OFFSET], TOP_OFFSET        ; 2615
              6E      0200    8F 3C 00191 4$:    MOVZWL   #512, STRING_DESC                         ; 2621
              50        F7    A3 9E 00196        MOVAB    -9(R3), R0                                ; 2623
              52              50 D1 0019A        CMPL     R0, TOP_OFFSET                            ;
                              17 15 0019D        BLEQ     5$                                        ;
              53              09 C2 0019F        SUBL2    #9, BOTTOM_OFFSET                         ; 2626
                              07 DD 001A2        PUSHL    #7                                        ; 2631
                              57 DD 001A4        PUSHL    R7                                        ; 2633
              53              57 C4 001A6        MULL2    R7, R3                                    ; 2631
```

```
                                      6344  9F 001A9          PUSHAB  (R3)[R4]                                    ;
                              0C        AE  9F 001AC          PUSHAB  STRING_DESC                                 ; 2630
                                        56  DD 001AF          PUSHL   R6                                          ; 2631
                    0000V  CF           05  FB 001B1          CALLS   #5, INSERT_JOBNUMBER_BANNER                 ;
                                            04 001B6 5$:      RET                                                 ; 2636

; Routine Size:  439 bytes,    Routine Base:  CODE + 098F
```

```
1 %sbttl 'FILL_JOB_TRAILER - Insert Information into the JOB Page'
1 !++
1 ! Functional Description:
1 !       This procedure controls all inserts required for the JOB Page.
1 !
1 ! Formal Parameters:
1 !       SCB             - Address of the SCB
1 !       PAGE_REF        - Pointer to the Page (first byte)
1 !       PAGE_LENGTH     - Length of Frame
1 !       PAGE_WIDTH      - Width of Frame
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1 ROUTINE FILL_JOB_TRAILER (
1       SCB             : REF $BBLOCK,
1       PAGE_REF        : REF PAGE_ARRAY,
1       PAGE_WIDTH,
1       PAGE_LENGTH
1       ):      NOVALUE =
2 BEGIN
2
2 LITERAL K_MAX_BUFFER_SIZE = 512;
2
2 LOCAL
2     RET_LEN     : VECTOR[1],
2     RET_WIDE    : VECTOR[1],
2     RIGHT_OFFSET,
2     LEFT_OFFSET ,
2     TOP_OFFSET  ,
2     BOTTOM_OFFSET,
2     BUFFER      : VECTOR [512,byte],             ! Assume max size 512 bytes
2     STRING_DESC : VECTOR [2];                    ! Descriptor to current string
2
2 ! Allocate the buffer for "GET_xxx" Routines
2 !
2 STRING_DESC[SIZE] = %ALLOCATION(BUFFER);         ! allocate for routines
2 STRING_DESC[ADDR] = BUFFER;                      ! init address
2
2 ! Top of page
2 !
2 TOP_OFFSET = 0;
2 BOTTOM_OFFSET = .PAGE_LENGTH;
2
2     !  insert the burst characters
2     FILL_FRAME (.SCB,
2               .SCB[PSM$B_JOB_BURST_CHAR],
2               PAGE_REF[0,.TOP_OFFSET+2,.PAGE_WIDTH], .PAGE_WIDTH, 3);
```

```

    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;        ! reset buffer size

    GET_EOJ
                 (.SCB,
                 STRING_DESC[0],                  ! Buffer descriptor
                 STRING_DESC[SIZE]);              ! Returned length

    RET_LEN[0] = INSERT_NAME_BANNER (
                         .SCB,
                         STRING_DESC[SIZE],       ! eoj name desc
                         PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
                                                  ! ref to frame
                         .PAGE_WIDTH,             ! max width Bann
                         .BOTTOM_OFFSET - .TOP_OFFSET,
                                                  ! frame size
                         7);                      ! max hght Bann str

    IF .RET_LEN[0] GTR 0
    THEN
        TOP_OFFSET       = .TOP_OFFSET + .RET_LEN[0] + 2;
                                                  ! adjust & allow for spacing
                                                  !  includes the burst also...
                                                  !  two spaces...
 ! Bottom of page
 !
BOTTOM_OFFSET = .PAGE_LENGTH - 5;                 ! offset includes burst offset

    FILL_FRAME (.SCB,
                 .SCB[PSM$B_JOB_BURST_CHAR],
                 PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH,3);

    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;        ! reset buffer size

    !  Get the digital logo and output to page ... assume not greater than
    !  amount allocated.... truncation otherwise occurs

    GET_DIGITAL_LOGO
                 (.SCB,                           ! SCB addr.
                 STRING_DESC[0],                  ! Buffer descriptor
                 STRING_DESC[SIZE]);              ! Returned length

    CENTER_FRAME (.SCB,
                    STRING_DESC[0],
                    PAGE_REF[0,.BOTTOM_OFFSET+1,.PAGE_WIDTH], .PAGE_WIDTH, 1);


    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;        ! reset buffer size

    GET_RULER_COARSE
                 (.SCB,                           ! SCB addr.
                 STRING_DESC[0],                  ! Buffer descriptor
                 STRING_DESC[SIZE]);              ! Returned length

```

```
; 1820     2751  2         SCROLL_FRAME (.SCB,
; 1821     2752  2                       STRING_DESC[0],
; 1822     2753  2                       PAGE_REF[0,.BOTTOM_OFFSET+4,.PAGE_WIDTH], .PAGE_WIDTH, 1);
; 1823     2754  2
; 1824     2755  2         SCROLL_FRAME (.SCB,
; 1825     2756  2                       $DESCRIPTOR ('1234567890'),
; 1826     2757  2                       PAGE_REF[0,.BOTTOM_OFFSET+5,.PAGE_WIDTH], .PAGE_WIDTH, 1);
; 1827     2758  2
; 1828     2759  2 !  Create a sentence describing the current job.
; 1829     2760  2 !
; 1830     2761  2         ! re-init
; 1831     2762  2         STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
; 1832     2763  2
; 1833     2764  2         GET_JOB_DESCRIPTION
; 1834     2765  2                     (.SCB,                            ! SCB addr.
; 1835     2766  2                      0,                               ! Use past tense
; 1836     2767  2                      STRING_DESC[0],                  ! Buffer descriptor
; 1837     2768  2                      STRING_DESC[SIZE]);              ! Returned length
; 1838     2769  2
; 1839     2770  2         RET_LEN[0] = RETURN_FRAME_LENGTH
; 1840     2771  2                      (.SCB,
; 1841     2772  2                       STRING_DESC[0],                 ! string ref.
; 1842     2773  2                       PAGE_REF[0,0,.PAGE_WIDTH],      ! ref to frame
; 1843     2774  2                       .PAGE_WIDTH,                    ! cols to fill
; 1844     2775  2                       .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill
; 1845     2776  2
; 1846     2777  2         IF .RET_LEN[0] GTR 0
; 1847     2778  2         THEN
; 1848     2779  2             BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
; 1849     2780  2                                                     ! adjust & allow for spacing
; 1850     2781  2                                                     !   before inserting
; 1851     2782  2         ! insert the string delimited
; 1852     2783  2         INSERT_FRAME (.SCB,
; 1853     2784  2                       STRING_DESC[0],                 ! string ref.
; 1854     2785  2                       PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH],
; 1855     2786  2                                                     ! ref to frame
; 1856     2787  2                       .PAGE_WIDTH,                    ! cols to fill
; 1857     2788  2                       .RET_LEN[0]);                   ! rows to fill
; 1858     2789  2
; 1859     2790  2
; 1860     2791  2 ! User name
; 1861     2792  2 !
; 1862     2793  2         RET_LEN[0] = INSERT_NAME_BANNER
; 1863     2794  2                           (.SCB,
; 1864     2795  2                           SCB_SIZE (USER_NAME),    ! user name desc
; 1865     2796  2                           PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 1866     2797  2                                                     ! ref to frame
; 1867     2798  2                           .PAGE_WIDTH,             ! max width Bann
; 1868     2799  2                           .BOTTOM_OFFSET - .TOP_OFFSET,
; 1869     2800  2                                                     ! frame size
; 1870     2801  2                           7);                      ! max hght Bann str desired
; 1871     2802  2
; 1872     2803  2         ! re-init
; 1873     2804  2         IF .RET_LEN[0] GTR 0
; 1874     2805  2         THEN
; 1875     2806  2             TOP_OFFSET       = .TOP_OFFSET + .RET_LEN[0] + 2;
; 1876     2807  2                                                     ! adjust & allow for spacing
```

```
2 ! Job name
2 !
2     ! re-init
2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
2
2     GET_JOB_NAME
2                 (.SCB,
2                 STRING_DESC[0],                  ! Buffer descriptor
2                 STRING_DESC[SIZE]);              ! Returned length
2
2     RET_LEN[0] = INSERT_NAME_BANNER (
2                         .SCB,
2                         STRING_DESC[SIZE],       ! job name size
2                         PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
2                                                  ! ref to frame
2                         .PAGE_WIDTH,             ! max width Bann
2                         .BOTTOM_OFFSET - .TOP_OFFSET,
2                                                  ! frame size
2                         7);                      ! max hght Bann str
2
2     ! re-init
2     IF .RET_LEN[0] GTR 0
2     THEN
2         TOP_OFFSET       = .TOP_OFFSET + .RET_LEN[0] + 2;
2                                                  ! adjust & allow for spacing
2 !  Receipt box
2 !
2 RIGHT_OFFSET = .PAGE_WIDTH;                      ! right_offset and left_offset
2 LEFT_OFFSET = 0;                                 ! are positional offsets for
2                                                  ! a specific frame_length and
2                                                  ! range.
2     ! re-init
2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
2
2     GET_RECEIPT_BOX
2                 (.SCB,                           ! SCB addr.
2                 STRING_DESC[0],                  ! Buffer descriptor
2                 STRING_DESC[SIZE]);              ! Returned length
2
2     ! Get the width needed for insert (assume length of seven)
2     RET_WIDE[0] = RETURN_FRAME_WIDTH
2                  (.SCB,
2                   STRING_DESC[0],                ! string ref.
2                   PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
2                                                  ! ref to frame
2                   .RIGHT_OFFSET -.LEFT_OFFSET,   ! special width
2                   8);                            ! rows to fill
2
2     RET_LEN[0] = RETURN_FRAME_LENGTH
2                  (.SCB,
2                   STRING_DESC[0],                ! string ref.
2                   PAGE_REF[0,0,.PAGE_WIDTH],     ! ref to frame
2                   .RET_WIDE[0],                  ! cols to fill
2                   .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill

2     IF .RET_LEN[0] GTR 0
2     THEN
```

```
; 1934      2865  2         BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
; 1935      2866  2                                                 ! adjust & allow for spacing
; 1936      2867  2                                                 !    before inserting
; 1937      2868  2     RIGHT_OFFSET = .RIGHT_OFFSET - .RET_WIDE[0];! offset before inserting
; 1938      2869  2
; 1939      2870  2     MOVE_FRAME (.SCB,
; 1940      2871  2                 STRING_DESC[0],                 ! string frame reference
; 1941      2872  2                 PAGE_REF[.RIGHT_OFFSET,.BOTTOM_OFFSET,.PAGE_WIDTH],
; 1942      2873  2                                                 ! ref to frame
; 1943      2874  2                 .RET_WIDE[0],                   ! width
; 1944      2875  2                 .RET_LEN[0]);                   ! rows to fill
; 1945      2876  2
; 1946      2877  2     !  Get and insert the filename banner
; 1947      2878  2     !
; 1948      2879  2
; 1949      2880  2     ! re-init
; 1950      2881  2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;       ! reset buffer size
; 1951      2882  2
; 1952      2883  2     INSERT_JOBNUMBER_BANNER
; 1953      2884  2                (.SCB,
; 1954      2885  2                 STRING_DESC[0],                 ! Buffer desc.
; 1955      2886  2                 PAGE_REF[.LEFT_OFFSET,.BOTTOM_OFFSET,.PAGE_WIDTH],
; 1956      2887  2                                                 ! ref to frame
; 1957      2888  2                 .RIGHT_OFFSET-.LEFT_OFFSET,     ! max width Bann
; 1958      2889  2                 .BOTTOM_OFFSET - .TOP_OFFSET);  ! rows to fill
; 1959      2890  2
; 1960      2891  2 ! re-init
; 1961      2892  2 STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;  ! reset buffer size
; 1962      2893  2 RIGHT_OFFSET = .PAGE_WIDTH;                     ! right_offset and left_offset
; 1963      2894  2 LEFT_OFFSET = 0;                                ! are positional offsets for
; 1964      2895  2                                                 ! a specific frame_length and
; 1965      2896  2                                                 ! range.
; 1966      2897  2
; 1967      2898  2 !     GET_ACCOUNTING_INFO
; 1968      2899  2 !               (.SCB,
; 1969      2900  2 !               STRING_DESC[0],         ! Buffer descriptor
; 1970      2901  2 !               STRING_DESC[SIZE]);     ! Returned length
; 1971      2902  2 !
; 1972      2903  2 ! insert the string delimited
; 1973      2904  2 !   INSERT_FRAME (.SCB,
; 1974      2905  2 !               STRING_DESC[0],                               ! string ref.
; 1975      2906  2 !               PAGE_REF[0, 45, .PAGE_WIDTH],                 ! ref to frame
; 1976      2907  2 !               .PAGE_WIDTH,                                  ! cols to fill
; 1977      2908  2 !               2);                                           ! rows to fill
; 1978      2909  1 END;


       30  39  38  37  36  35  34  33  32  31  00B46 P.AAF:  .ASCII   \1234567890\                          ;
                                     0000000A  00B50 P.AAE:  .LONG    10                                    ;
                                     00000000' 00B54         .ADDRESS P.AAF                                 ;


                                          0FFC 00000 FILL_JOB_TRAILER:
                                                             .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11  ; 2660
                     5B      0000V  CF   9E 00002            MOVAB    INSERT_NAME_BANNER, R11               ;
                     5E   FDFC   CE  9E 00007           MOVAB   -516(SP), SP                                  ;
                     7E   0200   8F  3C 0000C           MOVZWL  #512, STRING_DESC                             ; 2682
              04     AE     08   AE  9E 00011           MOVAB   BUFFER, STRING_DESC+4                         ; 2683
                                 52  D4 00016           CLRL    TOP_OFFSET                                    ; 2687
                     53     10   AC  D0 00018           MOVL    PAGE_LENGTH, BOTTOM_OFFSET                    ; 2688
                                 03  DD 0001C           PUSHL   #3                                            ; 2693
                     57     0C   AC  D0 0001E           MOVL    PAGE_WIDTH, R7                                ;
                                 57  DD 00022           PUSHL   R7                                            ;
                     54     08   AC  D0 00024           MOVL    PAGE_REF, R4                                  ;
                     50     02   A2  9E 00028           MOVAB   2(R2), R0                                     ;
                     50          57  C4 0002C           MULL2   R7, R0                                        ;
                               6044  9F 0002F           PUSHAB  (R0)[R4]                                      ;
                     56     04   AC  D0 00032           MOVL    SCB, R6                                       ; 2692
                     7E   02A6   C6  9A 00036           MOVZBL  678(R6), -(SP)                                ; 2693
                                 56  DD 0003B           PUSHL   R6                                            ;
           0000V     CF          05  FB 0003D           CALLS   #5, FILL_FRAME                                ;
                     6E   0200   8F  3C 00042           MOVZWL  #512, STRING_DESC                             ; 2696
                                 5E  DD 00047           PUSHL   SP                                            ; 2701
                            04   AE  9F 00049           PUSHAB  STRING_DESC                                   ; 2700
                                 56  DD 0004C           PUSHL   R6                                            ; 2699
           0000V     CF          03  FB 0004E           CALLS   #3, GET_EOJ                                   ;
                                 07  DD 00053           PUSHL   #7                                            ; 2706
       7E            53          52  C3 00055           SUBL3   TOP_CFFSET, BOTTOM_OFFSET, -(SP)              ; 2709
                                 57  DD 00059           PUSHL   R7                                            ; 2708
       50            52          57  C5 0005B           MULL3   R7, TOP_OFFSET, R0                            ; 2706
                               6044  9F 0005F           PUSHAB  (R0)[R4]                                      ;
                            10   AE  9F 00062           PUSHAB  STRING_DESC                                   ; 2705
                                 56  DD 00065           PUSHL   R6                                            ; 2706
                     6B          06  FB 00067           CALLS   #6, INSERT_NAME_BANNER                        ;
                     55          50  D0 0006A           MOVL    R0, RET_LEN                                   ;
                                 05  15 0006D           BLEQ    1$                                            ; 2713
                     52     02 A542  9E 0006F           MOVAB   2(RET_LEN)[TOP_OFFSET], TOP_OFFSET            ; 2715
       53     10     AC          05  C3 00074  1$:      SUBL3   #5, PAGE_LENGTH, BOTTOM_OFFSET                ; 2721
                                 03  DD 00079           PUSHL   #3                                            ; 2725
                                 57  DD 0007B           PUSHL   R7                                            ;
       50            53          57  C5 0007D           MULL3   R7, BOTTOM_OFFSET, R0                         ;
                               6044  9F 00081           PUSHAB  (R0)[R4]                                      ;
                     7E   02A6   C6  9A 00084           MOVZBL  678(R6), -(SP)                                ;
                                 56  DD 00089           PUSHL   R6                                            ;
           0000V     CF          05  FB 0008B           CALLS   #5, FILL_FRAME                                ;
                     6E   0200   8F  3C 00090           MOVZWL  #512, STRING_DESC                             ; 2728
                                 5E  DD 00095           PUSHL   SP                                            ; 2736
                            04   AE  9F 00097           PUSHAB  STRING_DESC                                   ; 2735
                                 56  DD 0009A           PUSHL   R6                                            ; 2734
           0000V     CF          03  FB 0009C           CALLS   #3, GET_DIGITAL_LOGO                          ;
                                 01  DD 000A1           PUSHL   #1                                            ; 2740
                                 57  DD 000A3           PUSHL   R7                                            ;
                     50     01   A3  9E 000A5           MOVAB   1(R3), R0                                     ;
                     50          57  C4 000A9           MULL2   R7, R0                                        ;
                               6044  9F 000AC           PUSHAB  (R0)[R4]                                      ;
                            0C   AE  9F 000AF           PUSHAB  STRING_DESC                                   ; 2739
                                 56  DD 000B2           PUSHL   R6                                            ; 2740
           0000V     CF          05  FB 000B4           CALLS   #5, CENTER_FRAME                              ;
                     6E   0200   8F  3C 000B9           MOVZWL  #512, STRING_DESC                             ; 2744
                                 5E  DD 000BE           PUSHL   SP                                            ; 2749
                            04   AE  9F 000C0           PUSHAB  STRING_DESC                                   ; 2748
                                 56  DD 000C3           PUSHL   R6                                            ; 2747
```

```
      0000V  CF              03  FB 000C5       CALLS    #3, GET_RULER_COARSE                         ;
                             01  DD 000CA       PUSHL    #1                                           ; 2753
                             57  DD 000CC       PUSHL    R7                                           ;
             50        04    A3  9E 000CE       MOVAB    4(R3), R0                                    ;
             50              57  C4 000D2       MULL2    R7, R0                                       ;
                           6044  9F 000D5       PUSHAB   (R0)[R4]                                     ;
                     0C      AE  9F 000D8       PUSHAB   STRING_DESC                                  ; 2752
                             56  DD 000DB       PUSHL    R6                                           ; 2753
      0000V  CF              05  FB 000DD       CALLS    #5, SCROLL_FRAME                             ;
                             01  DD 000E2       PUSHL    #1                                           ; 2757
                             57  DD 000E4       PUSHL    R7                                           ;
             50        05    A3  9E 000E6       MOVAB    5(R3), R0                                    ;
             50              57  C4 000EA       MULL2    R7, R0                                       ;
                           6044  9F 000ED       PUSHAB   (R0)[R4]                                     ;
                   FF04      CF  9F 000F0       PUSHAB   P.AAE                                        ; 2756
                             56  DD 000F4       PUSHL    R6                                           ; 2757
      0000V  CF              05  FB 000F6       CALLS    #5, SCROLL_FRAME                             ;
             6E    0200      8F  3C 000FB       MOVZWL   #512, STRING_DESC                            ; 2762
                             5E  DD 00100       PUSHL    SP                                           ; 2768
                       04    AE  9F 00102       PUSHAB   STRING_DESC                                  ; 2767
                             7E  D4 00105       CLRL     -(SP)                                        ; 2765
                             56  DD 00107       PUSHL    R6                                           ;
      0000V  CF              04  FB 00109       CALLS    #4, GET_JOB_DESCRIPTION                      ;
7E           53              52  C3 0010E       SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)             ; 2775
                   0090      8F  BB 00112       PUSHR    #^M<R4,R7>                                   ; 2773
                     0C      AE  9F 00116       PUSHAB   STRING_DESC                                  ; 2772
                             56  DD 00119       PUSHL    R6                                           ; 2773
      0000V  CF              05  FB 0011B       CALLS    #5, RETURN_FRAME_LENGTH                      ;
             55              50  D0 00120       MOVL     R0, RET_LEN                                  ;
                             08  15 00123       BLEQ     2$                                           ; 2777
50           53              55  C3 00125       SUBL3    RET_LEN, BOTTOM_OFFSET, R0                   ; 2779
             53        FF    A0  9E 00129       MOVAB    -1(R0), BOTTOM_OFFSET                        ;
                             55  DD 0012D 2$:   PUSHL    RET_LEN                                      ; 2788
                             57  DD 0012F       PUSHL    R7                                           ; 2787
50           53              57  C5 00131       MULL3    R7, BOTTOM_OFFSET, R0                        ; 2785
                           6044  9F 00135       PUSHAB   (R0)[R4]                                     ;
                     0C      AE  9F 00138       PUSHAB   STRING_DESC                                  ; 2784
                             56  DD 0013B       PUSHL    R6                                           ; 2785
      0000V  CF              05  FB 0013D       CALLS    #5, INSERT_FRAME                             ;
                             07  DD 00142       PUSHL    #7                                           ; 2796
7E           53              52  C3 00144       SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)             ; 2799
                             57  DD 00148       PUSHL    R7                                           ; 2798
50           52              57  C5 0014A       MULL3    R7, TOP_OFFSET, R0                           ; 2796
                           6044  9F 0014E       PUSHAB   (R0)[R4]                                     ;
                   016C      C6  9F 00151       PUSHAB   364(R6)                                      ; 2795
                             56  DD 00155       PUSHL    R6                                           ; 2796
             6B              06  FB 00157       CALLS    #6, INSERT_NAME_BANNER                       ;
             55              50  D0 0015A       MOVL     R0, RET_LEN                                  ;
                             05  15 0015D       BLEQ     3$                                           ; 2804
             52        02  A542  9E 0015F       MOVAB    2(RET_LEN)[TOP_OFFSET], TOP_OFFSET           ; 2806
             6E    0200      8F  3C 00164 3$:   MOVZWL   #512, STRING_DESC                            ; 2811
                             5E  DD 00169       PUSHL    SP                                           ; 2816
                       04    AE  9F 0016B       PUSHAB   STRING_DESC                                  ; 2815
                             56  DD 0016E       PUSHL    R6                                           ; 2814
      0000V  CF              03  FB 00170       CALLS    #3, GET_JOB_NAME                             ;
                             07  DD 00175       PUSHL    #7                                           ; 2821
7E           53              52  C3 00177       SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)             ; 2824
```

```
                                                57 DD 0017B          PUSHL    R7                                  ; 2823
              50            52                  57 C5 0017D          MULL3    R7, TOP_OFFSET, R0                  ; 2821
                                              6044 9F 00181          PUSHAB   (R0)[R4]
                                          10    AE 9F 00184          PUSHAB   STRING_DESC                         ; 2820
                                                56 DD 00187          PUSHL    R6                                  ; 2821
                            6B                  06 FB 00189          CALLS    #6, INSERT_NAME_BANNER
                            55                  50 D0 0018C          MOVL     R0, RET_LEN
                                                05 15 0018F          BLEQ     4$                                  ; 2829
                            52            02  A542 9E 00191          MOVAB    2(RET_LEN)[TOP_OFFSET], TOP_OFFSET  ; 2831
                            58                  57 D0 00196 4$:      MOVL     R7, RIGHT_OFFSET                    ; 2835
                                                59 D4 00199          CLRL     LEFT_OFFSET                         ; 2836
                            6E          0200    8F 3C 0019B          MOVZWL   #512, STRING_DESC                   ; 2840
                                                5E DD 001A0          PUSHL    SP                                  ; 2845
                                          04    AE 9F 001A2          PUSHAB   STRING_DESC                         ; 2844
                                                56 DD 001A5          PUSHL    R6                                  ; 2843
                         0000V  CF              03 FB 001A7          CALLS    #3, GET_RECEIPT_BOX
                                                08 DD 001AC          PUSHL    #8                                  ; 2851
              7E            58                  59 C3 001AE          SUBL3    LEFT_OFFSET, RIGHT_OFFSET, -(SP)    ; 2853
              50            52                  57 C5 001B2          MULL3    R7, TOP_OFFSET, R0                  ; 2851
                                              6044 9F 001B6          PUSHAB   (R0)[R4]
                                          0C    AE 9F 001B9          PUSHAB   STRING_DESC                         ; 2850
                                                56 DD 001BC          PUSHL    R6                                  ; 2851
                         0000V  CF              05 FB 001BE          CALLS    #5, RETURN_FRAME_WIDTH
                                5A              50 D0 001C3          MOVL     R0, RET_WIDE
              7E            53                  52 C3 001C6          SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)    ; 2861
                                        0410    8F BB 001CA          PUSHR    #^M<R4,R10>                         ; 2859
                                          0C    AE 9F 001CE          PUSHAB   STRING_DESC                         ; 2858
                                                56 DD 001D1          PUSHL    R6                                  ; 2859
                         0000V  CF              05 FB 001D3          CALLS    #5, RETURN_FRAME_LENGTH
                                55              50 D0 001D8          MOVL     R0, RET_LEN
                                                08 15 001DB          BLEQ     5$                                  ; 2863
              50            53                  55 C3 001DD          SUBL3    RET_LEN, BOTTOM_OFFSET, R0          ; 2865
                            53            FF    A0 9E 001E1          MOVAB    -1(R0), BOTTOM_OFFSET
                            58                  5A C2 001E5 5$:      SUBL2    RET_WIDE, RIGHT_OFFSET              ; 2868
                                                55 DD 001E8          PUSHL    RET_LEN                             ; 2875
                                                5A DD 001EA          PUSHL    RET_WIDE                            ; 2874
              55            53                  57 C5 001EC          MULL3    R7, BOTTOM_OFFSET, R5               ; 2872
              50            55                  58 C1 001F0          ADDL3    RIGHT_OFFSET, R5, R0
                                              6044 9F 001F4          PUSHAB   (R0)[R4]
                                          0C    AE 9F 001F7          PUSHAB   STRING_DESC                         ; 2871
                                                56 DD 001FA          PUSHL    R6                                  ; 2872
                         0000V  CF              05 FB 001FC          CALLS    #5, MOVE_FRAME
                                6E      0200    8F 3C 00201          MOVZWL   #512, STRING_DESC                   ; 2881
              7E            53                  52 C3 00206          SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)    ; 2889
              7E            58                  59 C3 0020A          SUBL3    LEFT_OFFSET, RIGHT_OFFSET, -(SP)    ; 2888
              50            55                  59 C1 0020E          ADDL3    LEFT_OFFSET, R5, R0                 ; 2886
                                              6044 9F 00212          PUSHAB   (R0)[R4]
                                          0C    AE 9F 00215          PUSHAB   STRING_DESC                         ; 2885
                                                56 DD 00218          PUSHL    R6                                  ; 2886
                         0000V  CF              05 FB 0021A          CALLS    #5, INSERT_JOBNUMBER_BANNER
                                6E      0200    8F 3C 0021F          MOVZWL   #512, STRING_DESC                   ; 2892
                                58              57 D0 00224          MOVL     R7, RIGHT_OFFSET                    ; 2893
                                                59 D4 00227          CLRL     LEFT_OFFSET                         ; 2894
                                                   04 00229          RET                                          ; 2909

; Routine Size:  554 bytes,    Routine Base:  CODE + 0B58
; 2
; 2
; 2
; 2
; 2
; 2
; 2
; 2
; 2
; 2
; 2
; 2
; 2

; R
```

```
1 %sbttl 'FILL_FILE_TRAILER - Insert Information into the FILE Page'
1 !++
1 ! Functional Description:
1 !       This procedure controls all inserts required for the FILE Page.
1 !
1 ! Formal Parameters:
1 !       SCB             - Address of the SCB
1 !       PAGE_REF        - Pointer to the Page (first byte)
1 !       PAGE_LENGTH     - Length of Frame
1 !       PAGE_WIDTH      - Width of Frame
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1 ROUTINE FILL_FILE_TRAILER (
1       SCB             : REF $BBLOCK,
1       PAGE_REF        : REF PAGE_ARRAY,
1       PAGE_WIDTH,
1       PAGE_LENGTH
1       ):      NOVALUE =
2 BEGIN
2
2 LITERAL K_MAX_BUFFER_SIZE = 512;
2
2 LOCAL
2     FORCE_LEN     ,
2     RET_LEN       : VECTOR[1],
2     TOP_OFFSET    ,
2     BOTTOM_OFFSET,
2     BUFFER        : VECTOR [512,byte],    ! Assume max size 512 bytes
2     STRING_DESC : VECTOR [2];             ! Descriptor to current string
2
2 ! Allocate the buffer for "GET_xxx" Routines
2 !
2 STRING_DESC[SIZE] = %ALLOCATION(BUFFER);          ! allocate for routines
2 STRING_DESC[ADDR] = BUFFER;                        ! init address
2
2
2 ! Top of page
2 !
2 TOP_OFFSET = 0;
2 BOTTOM_OFFSET = .PAGE_LENGTH;
2
2     FILL_FRAME (.SCB,
2                 .SCB[PSM$B_FILE_BURST_CHAR],
2                 PAGE_REF[0,.TOP_OFFSET+2,.PAGE_WIDTH], .PAGE_WIDTH, 3);
2
2     FILL_FRAME (.SCB,
                %C' ',
                PAGE_REF[10,..TOP_OFFSET+2,.PAGE_WIDTH], .PAGE_WIDTH-20, 3);

    FILL_FRAME (.SCB,
                .SCB[PSM$B_JOB_BURST_CHAR],
                PAGE_REF[13, .TOP_OFFSET+2,.PAGE_WIDTH], .PAGE_WIDTH-26, 3);

    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size

    GET_EOF     (.SCB,
                STRING_DESC[0],                 ! Buffer descriptor
                STRING_DESC[SIZE]);             ! Returned length

    RET_LEN[0] = INSERT_NAME_BANNER (
                        .SCB,
                        STRING_DESC[SIZE],      ! file name desc
                        PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
                                                ! ref to frame
                        .PAGE_WIDTH,            ! max width Bann
                        .BOTTOM_OFFSET - .TOP_OFFSET,
                                                ! frame size
                        7);                     ! max hght Bann str

    ! Adust for the burst characters too
    IF .RET_LEN[0] GTR 0
    THEN
        TOP_OFFSET      = .TOP_OFFSET + .RET_LEN[0] + 2;
                                                ! adjust & allow for spacing
                                                !   allow for two spaces...

! Bottom of page - Bottom_offset already adjusted
!
BOTTOM_OFFSET = .PAGE_LENGTH - 5;               ! offset includes burst offset

    FILL_FRAME (.SCB,
                .SCB[PSM$B_FILE_BURST_CHAR],
                PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH, 3);

    FILL_FRAME (.SCB,
                %C' ',
                PAGE_REF[10,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH-20, 3);

    FILL_FRAME (.SCB,                           ! Offset set... Add the diff
                .SCB[PSM$B_JOB_BURST_CHAR],
                PAGE_REF[14,.BOTTOM_OFFSET,.PAGE_WIDTH], .PAGE_WIDTH-28, 3);

    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size

    !  Get the sys$announce note and output to page
    GET_DIGITAL_LOGO
                (.SCB,                          ! SCB addr.
                STRING_DESC[0],                 ! Buffer descriptor
                STRING_DESC[SIZE]);             ! Returned length

    ! assume string will not over run the area... fail_safe is truncation
    CENTER_FRAME (.SCB,
                   STRING_DESC[0],
                   PAGE_REF[0,.BOTTOM_OFFSET+1,.PAGE_WIDTH],
                   .PAGE_WIDTH, 1);


    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size

    GET_RULER_COARSE
                (.SCB,                          ! SCB addr.
                STRING_DESC[0],                 ! Buffer descriptor
                STRING_DESC[SIZE]);             ! Returned length

    SCROLL_FRAME (.SCB,
                   STRING_DESC[0],
                   PAGE_REF[0,.BOTTOM_OFFSET + 4,.PAGE_WIDTH], .PAGE_WIDTH, 1);

    SCROLL_FRAME (.SCB,
                  $DESCRIPTOR ('1234567890'),
                   PAGE_REF[0,.BOTTOM_OFFSET + 5,.PAGE_WIDTH], .PAGE_WIDTH, 1);

!  Create a sentence describing the current job.
!
    ! re-init
    STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size

    GET_JOB_DESCRIPTION
                (.SCB,                          ! SCB addr.
                0,                              ! Use past tense
                STRING_DESC[0],                 ! Buffer descriptor
                STRING_DESC[SIZE]);             ! Returned length

    RET_LEN[0] = RETURN_FRAME_LENGTH
                 (.SCB,
                  STRING_DESC[0],               ! string ref.
                  PAGE_REF[0,0,.PAGE_WIDTH],    ! ref to frame
                  .PAGE_WIDTH,                  ! cols to fill
                  .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill

    IF .RET_LEN[0] GTR 0
    THEN
        BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
                                                ! adjust & allow for spacing
                                                !   before inserting
    ! insert the string delimited
    INSERT_FRAME (.SCB,
                  STRING_DESC[0],               ! string ref.
                  PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH],
                                                ! ref to frame
                  .PAGE_WIDTH,                  ! cols to fill
                  .RET_LEN[0]);                 ! rows to fill


!  Create a sentence describing the current file.
!
    ! re-init
```

```
2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size
2
2     GET_FILE_DESCRIPTION
2                (.SCB,                           ! SCB addr.
2                 STRING_DESC[0],                 ! Buffer descriptor
2                 STRING_DESC[SIZE]);             ! Returned length
2
2     RET_LEN[0] = RETURN_FRAME_LENGTH
2                (.SCB,
2                 STRING_DESC[0],                 ! string ref.
2                 PAGE_REF[0,0,.PAGE_WIDTH],      ! ref to frame
2                 .PAGE_WIDTH,                    ! cols to fill
2                 .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill
2
2     IF .RET_LEN[0] GTR 0
2     THEN
2         BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
2                                                 ! adjust & allow for spacing
2                                                 !   before inserting
2     ! insert the string delimited
2     INSERT_FRAME (.SCB,
2                 STRING_DESC[0],                 ! string ref.
2                 PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH],
2                                                 ! ref to frame
2                 .PAGE_WIDTH,                    ! cols to fill
2                 .RET_LEN[0]);                   ! rows to fill
2
2 ! User name banner
2 !
2     RET_LEN[0] = INSERT_NAME_BANNER (
2                        .SCB,
2                        SCB_SIZE (USER_NAME),    ! user name desc
2                        PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
2                                                 ! ref to frame
2                        .PAGE_WIDTH,             ! max width Bann
2                        .BOTTOM_OFFSET - .TOP_OFFSET,
2                                                 ! frame size
2                        7);                      ! max hght Bann str
2
2     IF .RET_LEN[0] GTR 0
2     THEN
2         TOP_OFFSET     = .TOP_OFFSET + .RET_LEN[0] + 2;
2                                                 ! adjust & allow for spacing
2
2 !  Get and insert the filename banner ... force the banner to be small (always)
2 !
2     ! re-init
2     STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size
2
2     FORCE_LEN = 7;
2     IF .BOTTOM_OFFSET - .TOP_OFFSET LSS .FORCE_LEN
2     THEN
2         FORCE_LEN = .BOTTOM_OFFSET - .TOP_OFFSET;
2
2     RET_LEN[0] = INSERT_FILENAME_BANNER
2                       (.SCB,
2                        STRING_DESC[0],          ! file name size
; 2208          3138  2                             PAGE_REF[0,.TOP_OFFSET,.PAGE_WIDTH],
; 2209          3139  2                                                       ! ref to frame
; 2210          3140  2                             .PAGE_WIDTH,              ! max width Bann
; 2211          3141  2                             .FORCE_LEN);
; 2212          3142  2                                                       ! max hght Bann str
; 2213          3143  2         IF .RET_LEN[0] GTR 0
; 2214          3144  2         THEN
; 2215          3145  2             TOP_OFFSET      = .TOP_OFFSET + .RET_LEN[0] + 2;
; 2216          3146  2                                                       ! adjust & allow for spacing
; 2217          3147  2
; 2218          3148  2   !  Create a phrase which includes all the appropriate qualifiers
; 2219          3149  2   !  describing the current print and insert from the bottom without spacing.
; 2220          3150  2   !
; 2221          3151  2   ! re-init
; 2222          3152  2         STRING_DESC[SIZE] = K_MAX_BUFFER_SIZE;      ! reset buffer size
; 2223          3153  2
; 2224          3154  2         GET_QUALIFIERS
; 2225          3155  2                 (.SCB,                              ! SCB addr.
; 2226          3156  2                  STRING_DESC[0],                    ! Buffer descriptor
; 2227          3157  2                  STRING_DESC[SIZE]);                ! Returned length
; 2228          3158  2
; 2229          3159  2         RET_LEN[0] = RETURN_FRAME_LENGTH
; 2230          3160  2                  (.SCB,
; 2231          3161  2                   STRING_DESC[0],                   ! string ref.
; 2232          3162  2                   PAGE_REF[0,0,.PAGE_WIDTH],        ! ref to frame
; 2233          3163  2                   .PAGE_WIDTH-12,                   ! less twelve chars.
; 2234          3164  2                   .BOTTOM_OFFSET - .TOP_OFFSET);! rows to fill
; 2235          3165  2
; 2236          3166  2         IF .RET_LEN[0] GTR 0
; 2237          3167  2         THEN
; 2238          3168  3             BEGIN
; 2239          3169  3             BOTTOM_OFFSET = .BOTTOM_OFFSET - (.RET_LEN[0] + 1);
; 2240          3170  3                                                       ! adjust & allow for spacing
; 2241          3171  3                                                       !   before inserting
; 2242          3172  3
; 2243          3173  3             ! move the string undelimited
; 2244          3174  3             MOVE_FRAME (.SCB,
; 2245          3175  3                   $DESCRIPTOR ('Qualifiers: '),
; 2246          3176  3                   PAGE_REF[0,.BOTTOM_OFFSET,.PAGE_WIDTH],
; 2247          3177  3                                                       ! ref to frame
; 2248          3178  3                   .PAGE_WIDTH,                      ! cols to fill
; 2249          3179  3                   .RET_LEN[0]);                     ! rows to fill
; 2250          3180  2             END;
; 2251          3181  2
; 2252          3182  2         ! insert the string delimited
; 2253          3183  2         INSERT_FRAME (.SCB,
; 2254          3184  2                   STRING_DESC[0],                   ! string ref.
; 2255          3185  2                   PAGE_REF[12,.BOTTOM_OFFSET,.PAGE_WIDTH],
; 2256          3186  2                                                       ! ref to frame
; 2257          3187  2                   .PAGE_WIDTH-12,                   ! cols to fill
; 2258          3188  2                   .RET_LEN[0]);                     ! rows to fill
; 2259          3189  2
; 2260          3190  1 END;



             30  39  38  37  36  35  34  33  32  31  00D82 P.AAH:  .ASCII  \1234567890\                          ;
```

```
                                      0000000A  00D8C P.AAG:   .LONG    10                                      ;
                                      00000000' 00D90          .ADDRESS P.AAH                                   ;
20  3A  73  72  65  69  66  69  6C  61  75  51  00D94 P.AAJ:   .ASCII   \Qualifiers: \                          ;
                                      0000000C  00DA0 P.AAI:   .LONG    12                                      ;
                                      00000000' 00DA4          .ADDRESS P.AAJ                                   ;



                                          0FFC  00000 FILL_FILE_TRAILER:
                                                               .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11    ; 2933
                          5B      0000V CF  9E  00002          MOVAB    INSERT_FRAME, R11                       ;
                          5A      0000V CF  9E  00007          MOVAB    RETURN_FRAME_LENGTH, R10                ;
                          59      0000V CF  9E  0000C          MOVAB    FILL_FRAME, R9                          ;
                          5E       FDFC CE  9E  00011          MOVAB    -516(SP), SP                            ;
                          7E       0200 8F  3C  00016          MOVZWL   #512, STRING_DESC                       ; 2953
                     04   AE         08 AE  9E  0001B          MOVAB    BUFFER, STRING_DESC+4                   ; 2954
                                        52  D4  00020          CLRL     TOP_OFFSET                              ; 2959
                          53         10 AC  D0  00022          MOVL     PAGE_LENGTH, BOTTOM_OFFSET              ; 2960
                                        03  DD  00026          PUSHL    #3                                      ; 2964
                          56         0C AC  D0  00028          MOVL     PAGE_WIDTH, R6                          ;
                                        56  DD  0002C          PUSHL    R6                                      ;
                          54         08 AC  D0  0002E          MOVL     PAGE_REF, R4                            ;
                          55         02 A2  9E  00032          MOVAB    2(R2), R5                               ;
                          55            56  C4  00036          MULL2    R6, R5                                  ;
                                      6544  9F  00039          PUSHAB   (R5)[R4]                                ;
                          58         04 AC  D0  0003C          MOVL     SCB, R8                                 ; 2963
                          7E       02A4 C8  9A  00040          MOVZBL   676(R8), -(SP)                          ; 2964
                                        58  DD  00045          PUSHL    R8                                      ;
                          69            05  FB  00047          CALLS    #5, FILL_FRAME                          ;
                                        03  DD  0004A          PUSHL    #3                                      ; 2968
                                     EC A6  9F  0004C          PUSHAB   -20(R6)                                 ;
                                   0A A544  9F  0004F          PUSHAB   10(R5)[R4]                              ;
                                        20  DD  00053          PUSHL    #32                                     ;
                                        58  DD  00055          PUSHL    R8                                      ;
                          69            05  FB  00057          CALLS    #5, FILL_FRAME                          ;
                                        03  DD  0005A          PUSHL    #3                                      ; 2972
                                     E6 A6  9F  0005C          PUSHAB   -26(R6)                                 ;
                                   0D A544  9F  0005F          PUSHAB   13(R5)[R4]                              ;
                          7E       02A6 C8  9A  00063          MOVZBL   678(R8), -(SP)                          ;
                                        58  DD  00068          PUSHL    R8                                      ;
                          69            05  FB  0006A          CALLS    #5, FILL_FRAME                          ;
                          6E       0200 8F  3C  0006D          MOVZWL   #512, STRING_DESC                       ; 2975
                                        5E  DD  00072          PUSHL    SP                                      ; 2979
                                     04 AE  9F  00074          PUSHAB   STRING_DESC                             ; 2978
                                        58  DD  00077          PUSHL    R8                                      ; 2977
                   0000V  CF            03  FB  00079          CALLS    #3, GET_EOF                             ;
                                        07  DD  0007E          PUSHL    #7                                      ; 2984
           7E             53            52  C3  00080          SUBL3    TOP_OFFSET, BOTTOM_OFFSET, -(SP)        ; 2987
                                        56  DD  00084          PUSHL    R6                                      ; 2986
           55             52            56  C5  00086          MULL3    R6, TOP_OFFSET, R5                      ; 2984
                                      6544  9F  0008A          PUSHAB   (R5)[R4]                                ;
                                     10 AE  9F  0008D          PUSHAB   STRING_DESC                             ; 2983
                                        58  DD  00090          PUSHL    R8                                      ; 2984
                   0000V  CF            06  FB  00092          CALLS    #6, INSERT_NAME_BANNER                  ;
                          57            50  D0  00097          MOVL     R0, RET_LEN                             ;
                                        05  15  0009A          BLEQ     1$                                      ; 2992
                          52       02 A742  9E  0009C          MOVAB    2(RET_LEN)[TOP_OFFSET], TOP_OFFSET      ; 2994
```

```
            53       10    AC          05  C3 000A1 1$:     SUBL3   #5, PAGE_LENGTH, BOTTOM_OFFSET          ; 3000
                                       03  DD 000A6         PUSHL   #3                                      ; 3004
                                       56  DD 000A8         PUSHL   R6                                      ;
            55             53          56  C5 000AA         MULL3   R6, BOTTOM_OFFSET, R5                   ;
                                     6544  9F 000AE         PUSHAB  (R5)[R4]                                ;
                           7E   02A4   C8  9A 000B1         MOVZBL  676(R8), -(SP)                          ;
                                       58  DD 000B6         PUSHL   R8                                      ;
                           69          05  FB 000B8         CALLS   #5, FILL_FRAME                          ;
                                       03  DD 000BB         PUSHL   #3                                      ; 3008
                                    EC A6  9F 000BD         PUSHAB  -20(R6)                                 ;
                                  0A A544  9F 000C0         PUSHAB  10(R5)[R4]                              ;
                                       20  DD 000C4         PUSHL   #32                                     ;
                                       58  DD 000C6         PUSHL   R8                                      ;
                           69          05  FB 000C8         CALLS   #5, FILL_FRAME                          ;
                                       03  DD 000CB         PUSHL   #3                                      ; 3012
                                    E4 A6  9F 000CD         PUSHAB  -28(R6)                                 ;
                                  0E A544  9F 000D0         PUSHAB  14(R5)[R4]                              ;
                           7E   02A6   C8  9A 000D4         MOVZBL  678(R8), -(SP)                          ;
                                       58  DD 000D9         PUSHL   R8                                      ;
                           69          05  FB 000DB         CALLS   #5, FILL_FRAME                          ;
                           6E   0200   8F  3C 000DE         MOVZWL  #512, STRING_DESC                       ; 3015
                                       5E  DD 000E3         PUSHL   SP                                      ; 3021
                                 04    AE  9F 000E5         PUSHAB  STRING_DESC                             ; 3020
                                       58  DD 000E8         PUSHL   R8                                      ; 3019
                0000V      CF          03  FB 000EA         CALLS   #3, GET_DIGITAL_LOGO                    ;
                                       01  DD 000EF         PUSHL   #1                                      ; 3026
                                       56  DD 000F1         PUSHL   R6                                      ; 3027
                           55    01    A3  9E 000F3         MOVAB   1(R3), R5                               ; 3026
                           55          56  C4 000F7         MULL2   R6, R5                                  ;
                                     6544  9F 000FA         PUSHAB  (R5)[R4]                                ;
                                 0C    AE  9F 000FD         PUSHAB  STRING_DESC                             ; 3025
                                       58  DD 00100         PUSHL   R8                                      ; 3026
                0000V      CF          05  FB 00102         CALLS   #5, CENTER_FRAME                        ;
                           6E   0200   8F  3C 00107         MOVZWL  #512, STRING_DESC                       ; 3031
                                       5E  DD 0010C         PUSHL   SP                                      ; 3036
                                 04    AE  9F 0010E         PUSHAB  STRING_DESC                             ; 3035
                                       58  DD 00111         PUSHL   R8                                      ; 3034
                0000V      CF          03  FB 00113         CALLS   #3, GET_RULER_COARSE                    ;
                                       01  DD 00118         PUSHL   #1                                      ; 3040
                                       56  DD 0011A         PUSHL   R6                                      ;
                           55    04    A3  9E 0011C         MOVAB   4(R3), R5                               ;
                           55          56  C4 00120         MULL2   R6, R5                                  ;
                                     6544  9F 00123         PUSHAB  (R5)[R4]                                ;
                                 0C    AE  9F 00126         PUSHAB  STRING_DESC                             ; 3039
                                       58  DD 00129         PUSHL   R8                                      ; 3040
                0000V      CF          05  FB 0012B         CALLS   #5, SCROLL_FRAME                        ;
                                       01  DD 00130         PUSHL   #1                                      ; 3044
                                       56  DD 00132         PUSHL   R6                                      ;
                           55    05    A3  9E 00134         MOVAB   5(R3), R5                               ;
                           55          56  C4 00138         MULL2   R6, R5                                  ;
                                     6544  9F 0013B         PUSHAB  (R5)[R4]                                ;
                               FEA2    CF  9F 0013E         PUSHAB  P.AAG                                   ; 3043
                                       58  DD 00142         PUSHL   R8                                      ; 3044
                0000V      CF          05  FB 00144         CALLS   #5, SCROLL_FRAME                        ;
                           6E   0200   8F  3C 00149         MOVZWL  #512, STRING_DESC                       ; 3049
                                       5E  DD 0014E         PUSHL   SP                                      ; 3055
                                 04    AE  9F 00150         PUSHAB  STRING_DESC                             ; 3054
```

```
                                   7E D4 00153           CLRL    -(SP)                                   ; 3052
                                   58 DD 00155           PUSHL   R8                                      ;
                0000V    CF        04 FB 00157           CALLS   #4, GET_JOB_DESCRIPTION                 ;
    7E                   53        52 C3 0015C           SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)        ; 3062
                              0050 8F BB 00160           PUSHR   #^M<R4,R6>                              ; 3060
                                0C AE 9F 00164           PUSHAB  STRING_DESC                             ; 3059
                                   58 DD 00167           PUSHL   R8                                      ; 3060
                         6A        05 FB 00169           CALLS   #5, RETURN_FRAME_LENGTH                 ;
                         57        50 D0 0016C           MOVL    R0, RET_LEN                             ;
                                   08 15 0016F           BLEQ    2$                                      ; 3064
    50                   53        57 C3 00171           SUBL3   RET_LEN, BOTTOM_OFFSET, R0              ; 3066
                         53     FF A0 9E 00175           MOVAB   -1(R0), BOTTOM_OFFSET                   ;
                         7E        56 7D 00179 2$:       MOVQ    R6, -(SP)                               ; 3074
    50                   53        56 C5 0017C           MULL3   R6, BOTTOM_OFFSET, R0                   ; 3072
                                 6044 9F 00180           PUSHAB  (R0)[R4]                                ;
                                0C AE 9F 00183           PUSHAB  STRING_DESC                             ; 3071
                                   58 DD 00186           PUSHL   R8                                      ; 3072
                         6B        05 FB 00188           CALLS   #5, INSERT_FRAME                        ;
                         6E   0200 8F 3C 0018B           MOVZWL  #512, STRING_DESC                       ; 3081
                                   5E DD 00190           PUSHL   SP                                      ; 3086
                                04 AE 9F 00192           PUSHAB  STRING_DESC                             ; 3085
                                   58 DD 00195           PUSHL   R8                                      ; 3084
                0000V    CF        03 FB 00197           CALLS   #3, GET_FILE_DESCRIPTION                ;
    7E                   53        52 C3 0019C           SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)        ; 3093
                              0050 8F BB 001A0           PUSHR   #^M<R4,R6>                              ; 3091
                                0C AE 9F 001A4           PUSHAB  STRING_DESC                             ; 3090
                                   58 DD 001A7           PUSHL   R8                                      ; 3091
                         6A        05 FB 001A9           CALLS   #5, RETURN_FRAME_LENGTH                 ;
                         57        50 D0 001AC           MOVL    R0, RET_LEN                             ;
                                   08 15 001AF           BLEQ    3$                                      ; 3095
    50                   53        57 C3 001B1           SUBL3   RET_LEN, BOTTOM_OFFSET, R0              ; 3097
                         53     FF A0 9E 001B5           MOVAB   -1(R0), BOTTOM_OFFSET                   ;
                         7E        56 7D 001B9 3$:       MOVQ    R6, -(SP)                               ; 3105
    50                   53        56 C5 001BC           MULL3   R6, BOTTOM_OFFSET, R0                   ; 3103
                                 6044 9F 001C0           PUSHAB  (R0)[R4]                                ;
                                0C AE 9F 001C3           PUSHAB  STRING_DESC                             ; 3102
                                   58 DD 001C6           PUSHL   R8                                      ; 3103
                         6B        05 FB 001C8           CALLS   #5, INSERT_FRAME                        ;
                                   07 DD 001CB           PUSHL   #7                                      ; 3113
    7E                   53        52 C3 001CD           SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)        ; 3116
                                   56 DD 001D1           PUSHL   R6                                      ; 3115
    55                   52        56 C5 001D3           MULL3   R6, TOP_OFFSET, R5                      ; 3113
                                 6544 9F 001D7           PUSHAB  (R5)[R4]                                ;
                              016C C8 9F 001DA           PUSHAB  364(R8)                                 ; 3112
                                   58 DD 001DE           PUSHL   R8                                      ; 3113
                0000V    CF        06 FB 001E0           CALLS   #6, INSERT_NAME_BANNER                  ;
                         57        50 D0 001E5           MOVL    R0, RET_LEN                             ;
                                   05 15 001E8           BLEQ    4$                                      ; 3120
                         52     02 A742 9E 001EA         MOVAB   2(RET_LEN)[TOP_OFFSET], TOP_OFFSET      ; 3122
                         6E   0200 8F 3C 001EF 4$:       MOVZWL  #512, STRING_DESC                       ; 3128
                         51        07 D0 001F4           MOVL    #7, FORCE_LEN                           ; 3130
    50                   53        52 C3 001F7           SUBL3   TOP_OFFSET, BOTTOM_OFFSET, R0           ; 3131
                         51        50 D1 001FB           CMPL    R0, FORCE_LEN                           ;
                                   03 18 001FE           BGEQ    5$                                      ;
                         51        50 D0 00200           MOVL    R0, FORCE_LEN                           ; 3133
                                   51 DD 00203 5$:       PUSHL   FORCE_LEN                               ; 3141
                                   56 DD 00205           PUSHL   R6                                      ; 3140
```

```
        55            52           56 C5 00207       MULL3   R6, TOP_OFFSET, R5                        ; 3138
                                 6544 9F 0020B       PUSHAB  (R5)[R4]                                  ;
                               0C  AE 9F 0020E       PUSHAB  STRING_DESC                               ; 3137
                                   58 DD 00211       PUSHL   R8                                        ; 3138
              0000V   CF           05 FB 00213       CALLS   #5, INSERT_FILENAME_BANNER                ;
                      57           50 D0 00218       MOVL    R0, RET_LEN                               ;
                                   05 15 0021B       BLEQ    6$                                        ; 3143
                      52       02 A742 9E 0021D       MOVAB   2(RET_LEN)[TOP_OFFSET], TOP_OFFSET        ; 3145
                      6E     0200  8F 3C 00222 6$:   MOVZWL  #512, STRING_DESC                         ; 3152
                                   5E DD 00227       PUSHL   SP                                        ; 3157
                               04  AE 9F 00229       PUSHAB  STRING_DESC                               ; 3156
                                   58 DD 0022C       PUSHL   R8                                        ; 3155
              0000V   CF           03 FB 0022E       CALLS   #3, GET_QUALIFIERS                        ;
        7E            53           52 C3 00233       SUBL3   TOP_OFFSET, BOTTOM_OFFSET, -(SP)          ; 3164
                               F4  A6 9F 00237       PUSHAB  -12(R6)                                   ; 3163
                                   54 DD 0023A       PUSHL   R4                                        ; 3162
                               0C  AE 9F 0023C       PUSHAB  STRING_DESC                               ; 3161
                                   58 DD 0023F       PUSHL   R8                                        ; 3162
                      6A           05 FB 00241       CALLS   #5, RETURN_FRAME_LENGTH                   ;
                      57           50 D0 00244       MOVL    R0, RET_LEN                               ;
                                   1D 15 00247       BLEQ    7$                                        ; 3166
        50            53           57 C3 00249       SUBL3   RET_LEN, BOTTOM_OFFSET, R0                ; 3169
                      53       FF  A0 9E 0024D       MOVAB   -1(R0), BOTTOM_OFFSET                     ;
                      7E           56 7D 00251       MOVQ    R6, -(SP)                                 ; 3178
        50            53           56 C5 00254       MULL3   R6, BOTTOM_OFFSET, R0                     ; 3176
                                 6044 9F 00258       PUSHAB  (R0)[R4]                                  ;
                             FD99  CF 9F 0025B       PUSHAB  P.AAI                                     ; 3175
                                   58 DD 0025F       PUSHL   R8                                        ; 3176
              0000V   CF           05 FB 00261       CALLS   #5, MOVE_FRAME                            ;
                                   57 DD 00266 7$:   PUSHL   RET_LEN                                   ; 3188
                               F4  A6 9F 00268       PUSHAB  -12(R6)                                   ; 3187
                      53           56 C4 0026B       MULL2   R6, R3                                    ; 3185
                               0C A344 9F 0026E       PUSHAB  12(R3)[R4]                                ;
                               0C  AE 9F 00272       PUSHAB  STRING_DESC                               ; 3184
                                   58 DD 00275       PUSHL   R8                                        ; 3185
                      6B           05 FB 00277       CALLS   #5, INSERT_FRAME                          ;
                                      04 0027A       RET                                               ; 3190

; Routine Size:  635 bytes,    Routine Base:  CODE + 0DA8
; 2262          3191    1 %sbttl 'RETURN_FRAME_LENGTH - Returns the Frame Length for String Insertion'
; 2263          3192    1 !++
; 2264          3193    1 ! Functional Description:
; 2265          3194    1 !       Returns the frame length needed to insert the string into the page.
; 2266          3195    1 !       This routine checks the top_of_frame/bottom_of_frame offsets and
; 2267          3196    1 !       decides if the string will fit=
; 2268          3197    1 !       1) yes - return number of frame rows required for string to fit.
; 2269          3198    1 !       2) no  - return zero
; 2270          3199    1 !
; 2271          3200    1 !
; 2272          3201    1 ! Formal Parameters:
; 2273          3202    1 !       SCB             - Address of the SCB
; 2274          3203    1 !       STR DESC        - Descriptor of String to Insert
; 2275          3204    1 !       FRAME_PTR       - Address of first byte of Frame
; 2276          3205    1 !       FRAME_LENGTH    - Length of Frame
; 2277          3206    1 !       FRAME_WIDTH     - Width of Frame
; 2278          3207    1 !
; 2279          3208    1 !
; 2280          3209    1 ! Implicit Inputs:
; 2281          3210    1 !               none
; 2282          3211    1 !
; 2283          3212    1 ! Implicit Outputs:
; 2284          3213    1 !               none
; 2285          3214    1 !
; 2286          3215    1 ! Returned Value:
; 2287          3216    1 !               none
; 2288          3217    1 !
; 2289          3218    1 ! Side Effects:
; 2290          3219    1 !               none
; 2291          3220    1 !--
; 2292          3221    1 ROUTINE RETURN_FRAME_LENGTH (
; 2293          3222    1       SCB             : REF $BBLOCK,
; 2294          3223    1       STR DESC        : REF VECTOR[2],
; 2295          3224    1       FRAME_PTR       : REF PAGE_ARRAY,
; 2296          3225    1       FRAME_WIDTH     ,               ! Number of Columns
; 2297          3226    1       FRAME_LENGTH                    ! Number of Rows
; 2298          3227    1                       ) =
; 2299          3228    2 BEGIN
; 2300          3229    2
; 2301          3230    2 LOCAL MAX_CHARS;
; 2302          3231    2
; 2303          3232    2 ! don't even try if there is no frame left
; 2304          3233    2 IF (.FRAME_LENGTH LEQ 0) OR
; 2305          3234    2     (.FRAME_WIDTH LEQ 0) OR
; 2306          3235    3     (.STR_DESC[SIZE] EQL 0)
; 2307          3236    2 THEN
; 2308          3237    2     RETURN 0;
; 2309          3238    2
; 2310          3239    2 MAX_CHARS = .FRAME_WIDTH * .FRAME_LENGTH;
; 2311          3240    2
; 2312          3241    2 ! The boundary condition of string size of some multiple of frame width
; 2313          3242    2 !    can occur - add one less than the frame width to overcome this condition
; 2314          3243    2 !
; 2315          3244    2 IF .STR_DESC[SIZE] LEQ .MAX_CHARS
; 2316          3245    2 THEN
; 2317          3246    2     RETURN ((.STR_DESC[SIZE]+(.FRAME_WIDTH-1)) / .FRAME_WIDTH);
; 2318          3247    2
; 2319              3248  2 ! otherwise it just won't fit
; 2320              3249  2 RETURN 0;
; 2321              3250  2
; 2322              3251  1 END;



                                                      0000 00000 RETURN_FRAME_LENGTH:
                                                                          .WORD   Save nothing                                   ; 3221
                                         14    AC   D5 00002              TSTL    FRAME_LENGTH                                   ; 3233
                                               23   15 00005              BLEQ    1$                                              ;
                                         10    AC   D5 00007              TSTL    FRAME_WIDTH                                     ; 3234
                                               1E   15 0000A              BLEQ    1$                                              ;
                                         08    BC   D5 0000C              TSTL    @STR_DESC                                       ; 3235
                                               19   13 0000F              BEQL    1$                                              ;
                  50        10     AC    14    AC   C5 00011              MULL3   FRAME_LENGTH, FRAME_WIDTH, MAX_CHARS           ; 3239
                                   50    08    BC   D1 00017              CMPL    @STR_DESC, MAX_CHARS                            ; 3244
                                               0D   14 0001B              BGTR    1$                                              ;
                  50        08     BC    10    AC   C1 0001D              ADDL3   FRAME_WIDTH, @STR_DESC, R0                      ; 3246
                                               50   D7 00023              DECL    R0                                              ;
                                   50    10    AC   C6 00025              DIVL2   FRAME_WIDTH, R0                                 ;
                                                    04 00029              RET                                                     ;
                                               50   D4 0002A 1$:          CLRL    R0                                              ; 3251
                                                    04 0002C              RET                                                     ;

; Routine Size:  45 bytes,    Routine Base:  CODE + 1023
; 2324       3252  1 %sbttl 'RETURN_FRAME_WIDTH - Returns the Frame Length for String Insertion'
; 2325       3253  1 !++
; 2326       3254  1 ! Functional Description:
; 2327       3255  1 !        Returns the frame width needed to insert the required lengths into
; 2328       3256  1 !        the page.  This routine returns only a prescribed value and
; 2329       3257  1 !        decides if the string will fit -
; 2330       3258  1 !        Return value - frame_width
; 2331       3259  1 !        Return zero - only if no length or width of frame.
; 2332       3260  1 !        Assumes the FRAME_LENGTH is constant. (How many lengths are needed
; 2333       3261  1 !        to fit this string)
; 2334       3262  1 !
; 2335       3263  1 ! Formal Parameters:
; 2336       3264  1 !        SCB                 - Address of the SCB
; 2337       3265  1 !        STR_DESC            - Descriptor of String to Insert
; 2338       3266  1 !        FRAME_PTR           - Address of first byte of Frame
; 2339       3267  1 !        FRAME_LENGTH        - Length of Frame
; 2340       3268  1 !        FRAME_WIDTH         - Width of Frame
; 2341       3269  1 !
; 2342       3270  1 !
; 2343       3271  1 ! Implicit Inputs:
; 2344       3272  1 !                none
; 2345       3273  1 !
; 2346       3274  1 ! Implicit Outputs:
; 2347       3275  1 !                none
; 2348       3276  1 !
; 2349       3277  1 ! Returned Value:
; 2350       3278  1 !                none
; 2351       3279  1 !
; 2352       3280  1 ! Side Effects:
; 2353       3281  1 !                none
; 2354       3282  1 !--
; 2355       3283  1 ROUTINE RETURN_FRAME_WIDTH (
; 2356       3284  1         SCB                 : REF $BBLOCK,
; 2357       3285  1         STR_DESC            : REF VECTOR[2],
; 2358       3286  1         FRAME_PTR           : REF PAGE_ARRAY,
; 2359       3287  1         FRAME_WIDTH         ,                       ! Number of Columns
; 2360       3288  1         FRAME_LENGTH                                ! Number of Rows
; 2361       3289  1                         ) =
; 2362       3290  2 BEGIN
; 2363       3291  2
; 2364       3292  2 LOCAL
; 2365       3293  2     MAX_CHARS,
; 2366       3294  2     TEMP_WIDE;
; 2367       3295  2
; 2368       3296  2 ! don't even try if there is no frame left
; 2369       3297  2 IF (.FRAME_LENGTH LEQ 0) OR
; 2370       3298  2     (.FRAME_WIDTH LEQ 0) OR
; 2371       3299  3     (.STR_DESC[SIZE] EQL 0)
; 2372       3300  2 THEN
; 2373       3301  2     RETURN 0;
; 2374       3302  2
; 2375       3303  2 MAX_CHARS = .FRAME_WIDTH * .FRAME_LENGTH;
; 2376       3304  2 IF .MAX_CHARS LSS .STR_DESC[SIZE]
; 2377       3305  2 THEN
; 2378       3306  2     RETURN 0;                               ! string wont fit
; 2379       3307  2
; 2380       3308  2 ! The boundary condition of string size of some multiple of frame width
; 2381          3309  2 !   can occur - add one less than the frame width to overcome this condition
; 2382          3310  2 !
; 2383          3311  2 TEMP_WIDE = .STR_DESC[SIZE]/.FRAME_LENGTH;
; 2384          3312  2
; 2385          3313  2 IF .TEMP_WIDE LEQ .FRAME_WIDTH
; 2386          3314  2 THEN
; 2387          3315  2     RETURN (.TEMP_WIDE);                             ! return the value
; 2388          3316  2
; 2389          3317  2 ! otherwise it just won't fit
; 2390          3318  2 RETURN 0;
; 2391          3319  2
; 2392          3320  1 END;




                                                   0000 00000 RETURN_FRAME_WIDTH:
                                                                         .WORD   Save nothing                           ; 3283
                           51        14      AC   D0 00002               MOVL    FRAME_LENGTH, R1                       ; 3297
                                             20   15 00006               BLEQ    1$                                     ;
                                     10      AC   D5 00008               TSTL    FRAME_WIDTH                            ; 3298
                                             1B   15 0000B               BLEQ    1$                                     ;
                                     08      BC   D5 0000D               TSTL    @STR_DESC                              ; 3299
                                             16   13 00010               BEQL    1$                                     ;
                  50        10      AC            51   C5 00012          MULL3   R1, FRAME_WIDTH, MAX_CHARS             ; 3303
                            08      BC            50   D1 00017          CMPL    MAX_CHARS, @STR_DESC                   ; 3304
                                             0B   19 0001B               BLSS    1$                                     ;
                  50        08      BC            51   C7 0001D          DIVL3   R1, @STR_DESC, TEMP_WIDE               ; 3311
                            10      AC            50   D1 00022          CMPL    TEMP_WIDE, FRAME_WIDTH                 ; 3313
                                             02   15 00026               BLEQ    2$                                     ;
                                             50   D4 00028 1$:           CLRL    R0                                     ; 3320
                                                  04 0002A 2$:           RET                                            ;

; Routine Size:  43 bytes,    Routine Base:  CODE + 1050
; 2394          3321   1 %sbttl 'GET_REVISION_DATE - Get the revision date of current file'
; 2395          3322   1 !++
; 2396          3323   1 ! Functional Description:
; 2397          3324   1 !       This routine creates a phrase with DD-MMM-YYYY HH:MM describing
; 2398          3325   1 !       the revision date of the current file.  Returns zero if file
; 2399          3326   1 !       not open.
; 2400          3327   1 !
; 2401          3328   1 ! Formal Parameters:
; 2402          3329   1 !       SCB             - Address of the SCB
; 2403          3330   1 !       STR_DESC        - Desc of String to Return
; 2404          3331   1 !       RET_LEN         - Return length of Desc.
; 2405          3332   1 !
; 2406          3333   1 ! Implicit Inputs:
; 2407          3334   1 !               none
; 2408          3335   1 !
; 2409          3336   1 ! Implicit Outputs:
; 2410          3337   1 !               none
; 2411          3338   1 !
; 2412          3339   1 ! Returned Value:
; 2413          3340   1 !               none
; 2414          3341   1 !
; 2415          3342   1 ! Side Effects:
; 2416          3343   1 !               none
; 2417          3344   1 !--
; 2418          3345   1 ROUTINE GET_REVISION_DATE       (
; 2419          3346   1                 SCB             : REF $BBLOCK,          ! SCB
; 2420          3347   1                 STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 2421          3348   1                 RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 2422          3349   1                         )       : NOVALUE =
; 2423          3350   2 BEGIN
; 2424          3351   2 BIND
; 2425          3352   2     XABDAT = .SCB[PSM$A_XABDAT]: $BBLOCK,        ! - RMS date block
; 2426        P 3353   2     FORMAT = $DESCRIPTOR (
; 2427          3354   2         '!17%D' ),                               ! - revision date
; 2428          3355   2
; 2429          3356   2     NAM = .SCB[PSM$A_NAM]: REF $BBLOCK;
; 2430          3357   2
; 2431          3358   2 LOCAL
; 2432          3359   2     CURRENT_LEN : INITIAL (0);
; 2433          3360   2
; 2434          3361   2 IF FILE_OPEN(.SCB)
; 2435          3362   2 THEN
; 2436        P 3363   2     $FAO (
; 2437        P 3364   2         FORMAT,
; 2438        P 3365   2         CURRENT_LEN,
; 2439        P 3366   2         STR_DESC[0],
; 2440        P 3367   2         XABDAT[XAB$Q_RDT],
; 2441          3368   2         );
; 2442          3369   2
; 2443          3370   2 RET_LEN[0] = .CURRENT_LEN;
; 2444          3371   2
; 2445          3372   1 END;


                              44  25  37  31  21   0107B P.AAL:   .ASCII  \!17%D\                                          ;
                                        00000005   01080 P.AAK:   .LONG   5                                                ;
```

```
                                       00000000' 01084          .ADDRESS P.AAL                                  ;

                                                          FORMAT=            P.AAK


                                            0004 00000 GET_REVISION_DATE:
                                                                .WORD    Save R2                                ; 3345
                        50         04    AC  D0 00002           MOVL     SCB, R0                                ; 3352
                        52       0254    C0  D0 00006           MOVL     596(R0), R2                            ;
                                         7E  D4 0000B           CLRL     CURRENT_LEN                            ; 3356
                                         50  DD 0000D           PUSHL    R0                                     ; 3361
                 0000V  CF               01  FB 0000F           CALLS    #1, FILE_OPEN                          ;
                        13               50  E9 00014           BLBC     R0, 1$                                 ;
                                   0C    A2  9F 00017           PUSHAB   12(R2)                                 ; 3368
                                   08    AC  DD 0001A           PUSHL    STR_DESC                               ;
                                   08    AE  9F 0001D           PUSHAB   CURRENT_LEN                            ;
                                   D5    AF  9F 00020           PUSHAB   FORMAT                                 ;
             00000000G  00               04  FB 00023           CALLS    #4, SYS$FAO                            ;
                    0C  BC               6E  B0 0002A 1$:       MOVW     CURRENT_LEN, @RET_LEN                  ; 3370
                                             04 0002E           RET                                             ; 3372

; Routine Size:  47 bytes,    Routine Base:  CODE + 1088
; 2447        3373  1 %sbttl 'GET_SYSTEM_ANNOUNCEMENT- Create a Sentence Describing the Current Job'
; 2448        3374  1 !++
; 2449        3375  1 ! Functional Description:
; 2450        3376  1 !       This routine get the system annoucement. All allocation of buffers
; 2451        3377  1 !       handled by caller
; 2452        3378  1 !
; 2453        3379  1 ! Formal Parameters:
; 2454        3380  1 !       SCB             - Address of the SCB
; 2455        3381  1 !       STR_DESC        - Desc of String to Return
; 2456        3382  1 !       RET_LEN         - Return length of Desc.
; 2457        3383  1 !
; 2458        3384  1 ! Implicit Inputs:
; 2459        3385  1 !               none
; 2460        3386  1 !
; 2461        3387  1 ! Implicit Outputs:
; 2462        3388  1 !               none
; 2463        3389  1 !
; 2464        3390  1 ! Returned Value:
; 2465        3391  1 !               none
; 2466        3392  1 !
; 2467        3393  1 ! Side Effects:
; 2468        3394  1 !               none
; 2469        3395  1 !--
; 2470        3396  1 ROUTINE GET_SYSTEM_ANNOUNCEMENT (
; 2471        3397  1               SCB             : REF $BBLOCK,          ! SCB
; 2472        3398  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 2473        3399  1               RSL_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 2474        3400  1                       )       : NOVALUE =
; 2475        3401  2 BEGIN
; 2476        3402  2 BIND
; 2477      P 3403  2     DEFINED_ANNOUNCE = $DESCRIPTOR (
; 2478        3404  2               '!AF'   ),              ! - Defined announcement
; 2479        3405  2
; 2480      P 3406  2     ANNOUNCE = $DESCRIPTOR (
; 2481        3407  2               'PSM$ANNOUNCE' );       ! - system annoucement
; 2482        3408  2
; 2483        3409  2 LOCAL
; 2484        3410  2     FAO_DESC    : VECTOR[2],
; 2485        3411  2     BUFFER      : VECTOR[256,byte],
; 2486        3412  2     TEMP_LEN    : INITIAL (0),
; 2487        3413  2     STATUS      ;
; 2488        3414  2
; 2489        3415  2 FAO_DESC[SIZE] = %ALLOCATION(BUFFER);
; 2490        3416  2 FAO_DESC[ADDR] = BUFFER;
; 2491        3417  2
; 2492        3418  2 !STATUS = $TRNLNM(attr  =LNM$M_CASE_BLIND,
; 2493        3419  2 !             tabnam=%ASCID 'LNM$SYSTEM_TABLE',
; 2494        3420  2 !             lognam= ANNOUNCE,
; 2495        3421  2 !             rsllen= FAO_DESC[SIZE],
; 2496        3422  2 !             rslbuf= FAO_DESC[ADDR]);
; 2497        3423  2
; 2498      P 3424  2 STATUS = $TRNLOG(lognam= ANNOUNCE,
; 2499      P 3425  2               rslbuf= FAO_DESC,
; 2500        3426  2               rsllen= TEMP_LEN);
; 2501        3427  2
; 2502        3428  2 IF .STATUS
; 2503        3429  2 THEN                          ! Success - Normal, Buffer_overflow
; 2504          3430  2       !
; 2505          3431  2       ! check for command file pointer "@" sign or no-translation code
; 2506          3432  2       !
; 2507          3433  3       BEGIN
; 2508          3434  4       IF (.STATUS EQL SS$_NOTRAN)
; 2509          3435  3       THEN
; 2510          3436  3           TEMP_LEN = 0;
; 2511          3437  3       END
; 2512          3438  2 ELSE                                          ! Bad status - Badparam, Badlength,
; 2513          3439  2       TEMP_LEN = 0;                           !  Badtable, Notfound(badname), Baddepth
; 2514          3440  2
; 2515          3441  2 IF .TEMP_LEN EQL 0                          ! Get the default sys version #
; 2516          3442  2 THEN
; 2517          3443  2       ! Put in the Digital Logo
; 2518          3444  2       GET_DIGITAL_LOGO(.SCB,STR_DESC[0],RSL_LEN[0])
; 2519          3445  2 ELSE
; 2520        P 3446  2       $FAO (
; 2521        P 3447  2           DEFINED_ANNOUNCE,
; 2522        P 3448  2           RSL_LEN[0],
; 2523        P 3449  2           STR_DESC[0],
; 2524        P 3450  2           .TEMP_LEN,
; 2525          3451  2           .FAO_DESC[ADDR]);
; 2526          3452  2
; 2527          3453  1 END;


                                          46  41  21  010B7 P.AAN:   .ASCII   \!AF\                          ;
                                                      010BA          .BLKB    2
                                            00000003  010BC P.AAM:   .LONG    3                              ;
                                            00000000' 010C0          .ADDRESS P.AAN                          ;
            45  43  4E  55  4F  4E  4E  41  24  4D  53  50  010C4 P.AAP:   .ASCII   \PSM$ANNOUNCE\                 ;
                                            0000000C  010D0 P.AAO:   .LONG    12                             ;
                                            00000000' 010D4          .ADDRESS P.AAP                          ;

                                                      DEFINED_ANNOUNCE=   P.AAM
                                                      ANNOUNCE=           P.AAO
                                                                .EXTRN   SYS$TRNLOG

                                             0000 00000 GET_SYSTEM_ANNOUNCEMENT:
                                                                .WORD    Save nothing                   ; 3396
                            5E       FEF8   CE   9E 00002        MOVAB    -264(SP), SP
                                             7E   D4 00007        CLRL     TEMP_LEN                       ; 3401
                      F8    AD       0100   8F   3C 00009        MOVZWL   #256, FAO_DESC                 ; 3415
                      FC    AD         04   AE   9E 0000F        MOVAB    BUFFER, FAO_DESC+4             ; 3416
                                             7E   7C 00014        CLRQ     -(SP)                          ; 3426
                                             7E   D4 00016        CLRL     -(SP)                          ;
                                       F8   AD   9F 00018        PUSHAB   FAO_DESC                       ;
                                       10   AE   9F 0001B        PUSHAB   TEMP_LEN                       ;
                                       D7   AF   9F 0001E        PUSHAB   ANNOUNCE                       ;
                  00000000G   00              06   FB 00021        CALLS    #6, SYS$TRNLOG                 ;
                              09              50   E9 00028        BLBC     STATUS, 1$                     ; 3428
                  00000629    8F              50   D1 0002B        CMPL     STATUS, #1577                  ; 3434
                                             02   12 00032        BNEQ     2$                             ;
                                             6E   D4 00034 1$:    CLRL     TEMP_LEN                       ; 3439
                                             6E   D5 00036 2$:    TSTL     TEMP_LEN                       ; 3441
                                             0D   12 00038        BNEQ     3$                             ;
                  7E        08    AC   7D 0003A          MOVQ     STR_DESC, -(SP)                               ; 3444
                            04    AC   DD 0003E          PUSHL    SCB                                           ;
             0000V  CF                 03  FB 00041      CALLS    #3, GET_DIGITAL_LOGO                          ;
                                       04 00046          RET                                                    ;
                            FC    AD   DD 00047 3$:      PUSHL    FAO_DESC+4                                    ; 3451
                            04    AE   DD 0004A          PUSHL    TEMP_LEN                                      ;
                            08    AC   DD 0004D          PUSHL    STR_DESC                                      ;
                            0C    AC   DD 00050          PUSHL    RSL_LEN                                       ;
                            8E    AF   9F 00053          PUSHAB   DEFINED_ANNOUNCE                              ;
         00000000G  00                 05  FB 00056      CALLS    #5, SYS$FAO                                   ;
                                       04 0005D          RET                                                    ; 3453

; Routine Size:  94 bytes,    Routine Base:  CODE + 10D8
; 2529        3454  1 %sbttl 'GET_VMS_LOGO - Create a Phrase of VMS logo'
; 2530        3455  1 !++
; 2531        3456  1 ! Functional Description:
; 2532        3457  1 !       VAX/VMS Version Vx.x
; 2533        3458  1 !
; 2534        3459  1 ! Formal Parameters:
; 2535        3460  1 !       SCB             - Address of the SCB
; 2536        3461  1 !       STR_DESC        - Desc of String to Return
; 2537        3462  1 !       RET_LEN         - Return length of Desc.
; 2538        3463  1 !
; 2539        3464  1 ! Implicit Inputs:
; 2540        3465  1 !               none
; 2541        3466  1 !
; 2542        3467  1 ! Implicit Outputs:
; 2543        3468  1 !               none
; 2544        3469  1 !
; 2545        3470  1 ! Returned Value:
; 2546        3471  1 !               none
; 2547        3472  1 !
; 2548        3473  1 ! Side Effects:
; 2549        3474  1 !               none
; 2550        3475  1 !--
; 2551        3476  1 ROUTINE GET_VMS_LOGO (
; 2552        3477  1               SCB             : REF $BBLOCK,          ! SCB
; 2553        3478  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 2554        3479  1               RSL_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 2555        3480  1                       )       : NOVALUE =
; 2556        3481  2 BEGIN
; 2557        3482  2 BIND
; 2558        3483  2     TRAILING = 1,
; 2559        3484  2
; 2560      P 3485  2     DEFAULT = $DESCRIPTOR (
; 2561      P 3486  2                   'VAX/VMS  ',
; 2562      P 3487  2                   'VAX/VMS  ',
; 2563      P 3488  2                   'VAX/VMS  ',
; 2564      P 3489  2                   'VAX/VMS  ',
; 2565      P 3490  2                   'VAX/VMS  ',
; 2566      P 3491  2                   'VAX/VMS  ',
; 2567      P 3492  2                   'VAX/VMS  ',
; 2568      P 3493  2                   'VAX/VMS  ',
; 2569      P 3494  2                   'VAX/VMS  ',
; 2570      P 3495  2                   'VAX/VMS  ',
; 2571      P 3496  2                   'VAX/VMS  ',
; 2572      P 3497  2                   'VAX/VMS  ',
; 2573      P 3498  2                   'VAX/VMS  ',
; 2574      P 3499  2                   'VAX/VMS  ',
; 2575      P 3500  2                   'VAX/VMS  ',
; 2576      P 3501  2                   'VAX/VMS  ',
; 2577      P 3502  2                   'VAX/VMS  ',
; 2578      P 3503  2                   'VAX/VMS  ',
; 2579      P 3504  2                   'VAX/VMS  ',
; 2580      P 3505  2                   'VAX/VMS  ',
; 2581      P 3506  2                   'VAX/VMS  ',
; 2582      P 3507  2                   'VAX/VMS  ',
; 2583      P 3508  2                   'VAX/VMS  ',
; 2584      P 3509  2                   'VAX/VMS  ',
; 2585        3510  2                   'VAX/VMS' );
; 2586        3511  2
; 2587        3512  2 LOCAL
; 2588        3513  2      STR_PTR
; 2589        3514  2      STR_LEN      ;
; 2590        3515  2
; 2591        3516  2 IF .SCB[PSM$L_PAGE_WIDTH] LSS 20
; 2592        3517  2 THEN                                           ! no room for burst bar
; 2593        3518  3      BEGIN
; 2594        3519  3      RSL_LEN[0] = 0;
; 2595        3520  3      RETURN;
; 2596        3521  2      END;
; 2597        3522  2
; 2598      P 3523  2 $FAO (
; 2599      P 3524  2        DEFAULT,
; 2600      P 3525  2      RSL_LEN[0],
; 2601        3526  2      STR_DESC[0]);
; 2602        3527  2
; 2603        3528  2 RSL_LEN[0] = .SCB[PSM$L_PAGE_WIDTH] - 20;         ! set the page length
; 2604        3529  2                                                  !  largest less than 180
; 2605        3530  2 STR_PTR = CH$PTR(.STR_DESC[ADDR]+.RSL_LEN[0]);
; 2606        3531  2
; 2607        3532  2 WHILE CH$NEQ( 1, .STR_PTR, 1, CH$PTR(UPLIT(' ')))
; 2608        3533  2 DO                                             ! trim off chars until blanks
; 2609        3534  3      BEGIN
; 2610        3535  3      RSL_LEN[0] = .RSL_LEN[0] - 1;
; 2611        3536  3      STR_PTR = CH$PTR(.STR_DESC[ADDR]+.RSL_LEN[0]);
; 2612        3537  2      END;
; 2613        3538  2
; 2614        3539  1 END;



                    20 20 53 4D 56 2F 58 41 56 01136 P.AAR:  .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 0113F         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01148         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01151         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 0115A         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01163         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 0116C         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01175         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 0117E         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01187         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01190         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01199         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011A2         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011AB         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011B4         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011BD         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011C6         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011CF         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011D8         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011E1         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011EA         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011F3         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 011FC         .ASCII  \VAX/VMS  \                                ;
                    20 20 53 4D 56 2F 58 41 56 01205         .ASCII  \VAX/VMS  \                                ;
                          53 4D 56 2F 58 41 56 0120E         .ASCII  \VAX/VMS\                                  ;
```

```
                                      01215         .BLKB    3                                        ;
                             000000DF 01218 P.AAQ:  .LONG    223                                      ;
                            00000000' 0121C         .ADDRESS P.AAR                                    ;
                       00  00  00  20 01220 P.AAS:  .ASCII   \ \<0><0><0>                             ;

                                            TRAILING=        1
                                            DEFAULT=         P.AAQ


                                001C 00000 GET_VMS_LOGO:
                                                   .WORD    Save R2,R3,R4                             ; 3476
                   53         04 AC D0 00002       MOVL     SCB, R3                                   ; 3516
                   14       0200 C3 D1 00006       CMPL     512(R3), #20                              ;
                                 04 18 0000B       BGEQ     1$                                        ;
                              0C BC B4 0000D       CLRW     @RSL_LEN                                  ; 3519
                                    04 00010       RET                                                ; 3518
                   54         08 AC D0 00011 1$:   MOVL     STR_DESC, R4                              ; 3526
                                 54 DD 00015       PUSHL    R4                                        ;
                   52         0C AC D0 00017       MOVL     RSL_LEN, R2                               ;
                                 52 DD 0001B       PUSHL    R2                                        ;
                              D4 AF 9F 0001D       PUSHAB   DEFAULT                                   ;
           00000000G  00         03 FB 00020       CALLS    #3, SYS$FAO                               ;
          62    0200  C3         14 A3 00027       SUBW3    #20, 512(R3), (R2)                        ; 3528
                      50         62 3C 0002D 2$:   MOVZWL   (R2), STR_PTR                             ; 3530
                      50      04 A4 C0 00030       ADDL2    4(R4), STR_PTR                            ;
                 C4   AF         60 91 00034       CMPB     (STR_PTR), P.AAS                          ; 3532
                                 04 13 00038       BEQL     3$                                        ;
                                 62 B7 0003A       DECW     (R2)                                      ; 3535
                                 EF 11 0003C       BRB      2$                                        ; 3536
                                    04 0003E 3$:   RET                                                ; 3539

; Routine Size:  63 bytes,    Routine Base:  CODE + 1224
; 2616            3540   1 %sbttl 'GET_DIGITAL_LOGO - Create a Phrase of Digital logo'
; 2617            3541   1 !++
; 2618            3542   1 ! Functional Description:
; 2619            3543   1 !       VAX/VMS Version Vx.x
; 2620            3544   1 !
; 2621            3545   1 ! Formal Parameters:
; 2622            3546   1 !       SCB             - Address of the SCB
; 2623            3547   1 !       STR_DESC        - Desc of String to Return
; 2624            3548   1 !       RET_LEN         - Return length of Desc.
; 2625            3549   1 !
; 2626            3550   1 ! Implicit Inputs:
; 2627            3551   1 !               none
; 2628            3552   1 !
; 2629            3553   1 ! Implicit Outputs:
; 2630            3554   1 !               none
; 2631            3555   1 !
; 2632            3556   1 ! Returned Value:
; 2633            3557   1 !               none
; 2634            3558   1 !
; 2635            3559   1 ! Side Effects:
; 2636            3560   1 !               none
; 2637            3561   1 !--
; 2638            3562   1 ROUTINE GET_DIGITAL_LOGO (
; 2639            3563   1                 SCB             : REF $BBLOCK,          ! SCB
; 2640            3564   1                 STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 2641            3565   1                 RSL_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 2642            3566   1                         )       : NOVALUE =
; 2643            3567   2 BEGIN
; 2644            3568   2 BIND
; 2645          P 3569   2     DEFAULT = $DESCRIPTOR (
; 2646          P 3570   2                 '!AC - VAX/VMS Version ',
; 2647            3571   2                 '!AS');
; 2648            3572   2
; 2649            3573   2 LOCAL
; 2650            3574   2     LOGO        ,
; 2651            3575   2     FAO_DESC    : VECTOR[2],
; 2652            3576   2     BUFFER      : VECTOR[20,byte],
; 2653            3577   2     ITEM_LIST   : $ITMLST_DECL (ITEMS=1);
; 2654            3578   2
; 2655            3579   2 FAO_DESC[SIZE] = %ALLOCATION(BUFFER);
; 2656            3580   2 FAO_DESC[ADDR] = BUFFER;
; 2657            3581   2
; 2658            3582   2 IF .SCB[PSM$L_PAGE_WIDTH] LSS 52                ! 52 chars in complete logo
; 2659            3583   2 THEN
; 2660            3584   2     LOGO = UPLIT BYTE (%ASCIC 'DEC')
; 2661            3585   2 ELSE
; 2662            3586   2     LOGO = UPLIT BYTE (%ASCIC 'Digital Equipment Corporation');
; 2663            3587   2
; 2664          P 3588   2 $ITMLST_INIT (ITMLST=ITEM_LIST,
; 2665          P 3589   2     (
; 2666          P 3590   2     ITMCOD=SYI$_VERSION,
; 2667          P 3591   2     BUFADR=.FAO_DESC[ADDR],
; 2668          P 3592   2     BUFSIZ=8,
; 2669          P 3593   2     RETLEN=FAO_DESC[SIZE]
; 2670            3594   2     ));
; 2671            3595   2
; 2672            3596   2 $GETSYIW(ITMLST=ITEM_LIST);
; 2673            3597  2
; 2674            3598  2 FAO_DESC[SIZE] = DELIMIT_STRING_NOT (.FAO_DESC[ADDR],
; 2675            3599  2                                        %CHAR(32), .FAO_DESC[SIZE]);
; 2676          P 3600  2 $FAO (
; 2677          P 3601  2      DEFAULT,
; 2678          P 3602  2      RSL_LEN[0],
; 2679          P 3603  2      STR_DESC[0],
; 2680          P 3604  2      .LOGO,
; 2681            3605  3      FAO_DESC[0])                     ! version
; 2682            3606  3
; 2683            3607  1 END;



56 20 53 4D 56 2F 58 41 56 20 2D 20 43 41 21 01263 P.AAU:  .ASCII  \!AC - VAX/VMS Version \            ;
                        20 6E 6F 69 73 72 65 01272                                                      ;
                                    53 41 21 01279         .ASCII  \!AS\                                ;
                                 00000019    0127C P.AAT:  .LONG   25                                   ;
                                 00000000'   01280         .ADDRESS P.AAU                               ;
                                 43 45 44 03 01284 P.AAV:  .ASCII  <3>\DEC\                             ;
6D 70 69 75 71 45 20 6C 61 74 69 67 69 44 1D 01288 P.AAW:  .ASCII  <29>\Digital Equipment Corporation\  ;
6E 6F 69 74 61 72 6F 70 72 6F 43 20 74 6E 65 01297                                                      ;

                                                   DEFAULT=                P.AAT
                                                           .EXTRN  SYS$GETSYIW

                                        0004 00000 GET_DIGITAL_LOGO:
                                                           .WORD   Save R2                           ; 3562
                     5E              2C  C2 00002          SUBL2   #44, SP                           ;
                 24  AE              14  D0 00005          MOVL    #20, FAO_DESC                     ; 3579
                 28  AE          10  AE  9E 00009          MOVAB   BUFFER, FAO_DESC+4                ; 3580
                     50          04  AC  D0 0000E          MOVL    SCB, R0                           ; 3582
                     34        0200  C0  D1 00012          CMPL    512(R0), #52                      ;
                                     06  18 00017          BGEQ    1$                                ;
                     52          C2  AF  9E 00019          MOVAB   P.AAV, LOGO                       ; 3584
                                     04  11 0001D          BRB     2$                                ;
                     52          C0  AF  9E 0001F 1$:      MOVAB   P.AAW, LOGO                       ; 3586
                     50              6E  9E 00023 2$:      MOVAB   ITEM_LIST, $$ITMBLKPTR            ; 3594
                     80    10000008  8F  D0 00026          MOVL    #268435464, ($$ITMBLKPTR)+        ;
                     80          28  AE  D0 0002D          MOVL    FAO_DESC+4, ($$ITMBLKPTR)+        ;
                     80          24  AE  9E 00031          MOVAB   FAO_DESC, ($$ITMBLKPTR)+          ;
                                     80  D4 00035          CLRL    ($$ITMBLKPTR)+                    ;
                                     7E  7C 00037          CLRQ    -(SP)                             ; 3596
                                     7E  D4 00039          CLRL    -(SP)                             ;
                                 0C  AE  9F 0003B          PUSHAB  ITEM_LIST                         ;
                                     7E  7C 0003E          CLRQ    -(SP)                             ;
                                     7E  D4 00040          CLRL    -(SP)                             ;
          00000000G  00              07  FB 00042          CALLS   #7, SYS$GETSYIW                   ;
                                 24  AE  DD 00049          PUSHL   FAO_DESC                          ; 3599
                                     20  DD 0004C          PUSHL   #32                               ; 3598
                                 30  AE  DD 0004E          PUSHL   FAO_DESC+4                        ;
              0000V  CF              03  FB 00051          CALLS   #3, DELIMIT_STRING_NOT            ;
                 24  AE              50  D0 00056          MOVL    R0, FAO_DESC                      ;
                                 24  AE  9F 0005A          PUSHAB  FAO_DESC                          ; 3605
                                     52  DD 0005D          PUSHL   LOGO                              ;
                                 08  AC  DD 0005F          PUSHL   STR_DESC                          ;
                                 0C  AC  DD 00062          PUSHL   RSL_LEN                           ;
```

```
                                   FF6D    CF  9F 00065          PUSHAB  DEFAULT                                          ;
                       00000000G  00       05  FB 00069          CALLS   #5, SYS$FAO                                      ;
                                               04 00070          RET                                                      ; 3607

; Routine Size:  113 bytes,    Routine Base:  CODE + 12A6
; 2685        3608 1 %sbttl 'GET_JOB_DESCRIPTION - Create a Sentence Describing the Current Job'
; 2686        3609 1 !++
; 2687        3610 1 ! Functional Description:
; 2688        3611 1 !        This routine creates a sentence describing the current job.
; 2689        3612 1 !
; 2690        3613 1 ! Formal Parameters:
; 2691        3614 1 !        SCB             - Address of the SCB
; 2692        3615 1 !        STR_DESC        - Desc of String to Return
; 2693        3616 1 !        RET_LEN         - Return length of Desc.
; 2694        3617 1 !
; 2695        3618 1 ! Implicit Inputs:
; 2696        3619 1 !                none
; 2697        3620 1 !
; 2698        3621 1 ! Implicit Outputs:
; 2699        3622 1 !                none
; 2700        3623 1 !
; 2701        3624 1 ! Returned Value:
; 2702        3625 1 !                none
; 2703        3626 1 !
; 2704        3627 1 ! Side Effects:
; 2705        3628 1 !                none
; 2706        3629 1 !--
; 2707        3630 1 ROUTINE GET_JOB_DESCRIPTION (
; 2708        3631 1                 SCB                     : REF $BBLOCK,          ! SCB
; 2709        3632 1                 TIME_FLAG
                                   ,
; 2710        3633 1                 STR_DESC                : REF VECTOR[2],        ! Output buffer desc
; 2711        3634 1                 RET_LEN                 : REF VECTOR [,WORD]    ! Return length (word)
; 2712        3635 1                                 )       : NOVALUE =
; 2713        3636 2 BEGIN
; 2714        3637 2 BIND
; 2715        3638 2     TRAILING     = 1,
; 2716        3639 2     LEADING      = 0,
; 2717        3640 2
; 2718      P 3641 2     NODE = $DESCRIPTOR (
; 2719        3642 2         'SYS$NODE'),! - system annoucement
; 2720        3643 2
; 2721      P 3644 2     DATE_FORMAT = $DESCRIPTOR (
; 2722        3645 2         '!17%D'),
; 2723        3646 2
; 2724      P 3647 2     SENT_FORMAT1 = $DESCRIPTOR (
; 2725      P 3648 2         'Job !AS ',                             ! - job name
; 2726      P 3649 2         '(!UL) ',                               ! - job number
; 2727      P 3650 2         'queued to !AS ',                       ! - batch file name(pres tense)
; 2728      P 3651 2         'on !AS ',                              ! - time queued
; 2729      P 3652 2         'by user !AS, ',                        ! - user name
; 2730      P 3653 2         'UIC !%I, ',                            ! - user uic
; 2731      P 3654 2         'under account !AS ',                   ! - user account
; 2732      P 3655 2         'at priority !UL, ',                    ! - que priority
; 2733      P 3656 2         '!AC ',                                 ! - 'started'/'completed'/
; 2734      P 3657 2                                                 !   'restarted'/'aborted'
; 2735      P 3658 2         'on printer !AS ',                      ! - device name
; 2736      P 3659 2         'on !AS ',                              ! - time printed
; 2737      P 3660 2         'from queue !AS',                       ! - executor queue
; 2738        3661 2         '.'      );                             ! - period
; 2739        3662 2
; 2740        3663 2 LOCAL
; 2741        3664 2     RET_LENGTH          ,
2	    STATUS                   ,
2	    DOUBLE_COLONS            ,
2	    CHOICE                   ,
2	    DATE_QUEUED              : VECTOR[2],
2	    DATE_QUEUED_BUFF         : VECTOR [17,byte],
2	    DATE_PRINTED             : VECTOR[2],
2	    DATE_PRINTED_BUFF        : VECTOR [17,byte],
2	    ACCOUNT_DESC             : VECTOR [2],
2	    USERNAME_DESC            : VECTOR [2];    ! desc of string
2
2	  ! get the user name delimited
2	  !
2	USERNAME_DESC[SIZE] = .SCB_SIZE_ (USER_NAME);
2	USERNAME_DESC[ADDR] = .SCB_ADDR_ (USER_NAME);
2	  ! Insert only the string ... No trailing blanks
2	  !
2	DISCARD (TRAILING, %C' ', .USERNAME_DESC[ADDR], .USERNAME_DESC[SIZE],
2	         USERNAME_DESC[SIZE],USERNAME_DESC[ADDR]); !Return length and pointer
2
2	  ! get the account name delimited
2	  !
2	ACCOUNT_DESC[SIZE] = .SCB_SIZE_ (ACCOUNT_NAME);
2	ACCOUNT_DESC[ADDR] = .SCB_ADDR_ (ACCOUNT_NAME);
2	  ! Insert only the string ... No trailing blanks
2	  !
2	DISCARD (TRAILING, %C' ', .ACCOUNT_DESC[ADDR], .ACCOUNT_DESC[SIZE],
2	         ACCOUNT_DESC[SIZE],ACCOUNT_DESC[ADDR]); ! Return length and pointer
2
2	  ! start, restart and complete
2	IF .TIME_FLAG
2	THEN
3	    BEGIN
3	    CHOICE = UPLIT BYTE (%ASCIC 'started');
3	    IF .REQUEST_FLAG_ (RESTARTING)
3	    THEN
3	        CHOICE = UPLIT BYTE (%ASCIC 'restarted');
3	    END
2	ELSE
3	    BEGIN
3	    BIND CONDITION = SCB[PSM$T_CONDITION_AREA] : VECTOR; ! Task completion status
3
3	    ! Assume job completed normally
3	    !
3	    CHOICE = UPLIT BYTE (%ASCIC 'completed');
3
3	    ! Check completion status for an error
3	    !
3	    IF .CONDITION[0] NEQU 0
3	    THEN
4	        BEGIN
4	        ! Assume job controller or symbiont initiated abort
4	        !
4	        CHOICE = UPLIT BYTE (%ASCIC 'ABORTED');
4
4	        ! Check for special case of job controller inititated requeue
4	        !
4
```

```
; 2799           3722  4 !*! FEATURE DISABLED UNTIL JOB CONTROLLER MESSAGE AVAILABILTY STRAIGHTENED OUT
; 2800           3723  4 !*!        IF .CONDITION[0] EQLU JBC$_JOBREQUEUE
; 2801           3724  4 !*!        THEN
; 2802           3725  4 !*!            CHOICE = UPLIT BYTE (%ASCIC 'REQUEUED');
; 2803           3726  4
; 2804           3727  3            END;
; 2805           3728  2        END;
; 2806           3729  2
; 2807           3730  2 ! Get and delimit the date/times
; 2808           3731  2 !    time queued
; 2809           3732  2 DATE_QUEUED[SIZE] = %ALLOCATION(DATE_QUEUED_BUFF);
; 2810           3733  2 DATE_QUEUED[ADDR] = DATE_QUEUED_BUFF;
; 2811           3734  2
; 2812         P 3735  2 $FAO (  DATE_FORMAT,
; 2813         P 3736  2         RET_LENGTH,
; 2814         P 3737  2         DATE_QUEUED[0],
; 2815           3738  2         SCB[PSM$Q_TIME_QUEUED]);
; 2816           3739  2 !
; 2817           3740  2 DISCARD (LEADING, %C' ', .DATE_QUEUED[ADDR], .RET_LENGTH,
; 2818           3741  2           DATE_QUEUED[SIZE],DATE_QUEUED[ADDR]);  ! Return length and pointer
; 2819           3742  2 DISCARD (TRAILING, %C' ', .DATE_QUEUED[ADDR], .RET_LENGTH,
; 2820           3743  2           DATE_QUEUED[SIZE],DATE_QUEUED[ADDR]);  ! Return length and pointer
; 2821           3744  2
; 2822           3745  2 !    time printed
; 2823           3746  2 DATE_PRINTED[SIZE] = %ALLOCATION(DATE_PRINTED_BUFF);
; 2824           3747  2 DATE_PRINTED[ADDR] = DATE_PRINTED_BUFF;
; 2825           3748  2
; 2826         P 3749  2 $FAO (  DATE_FORMAT,
; 2827         P 3750  2         RET_LENGTH,
; 2828         P 3751  2         DATE_PRINTED[0],
; 2829           3752  2         SCB[PSM$Q_TIME_PRINTED]);
; 2830           3753  2 !
; 2831           3754  2 DISCARD (LEADING, %C' ', .DATE_PRINTED[ADDR], .RET_LENGTH,
; 2832           3755  2           DATE_PRINTED[SIZE],DATE_PRINTED[ADDR]);! Return length and pointer
; 2833           3756  2 DISCARD (TRAILING, %C' ', .DATE_PRINTED[ADDR], .RET_LENGTH,
; 2834           3757  2           DATE_PRINTED[SIZE],DATE_PRINTED[ADDR]);! Return length and pointer
; 2835           3758  2
; 2836         P 3759  2 $FAO (  SENT_FORMAT1,
; 2837         P 3760  2         RET_LEN[0],
; 2838         P 3761  2         STR_DESC[0],
; 2839         P 3762  2         SCB[PSM$Q_JOB_NAME],                          ! job name
; 2840         P 3763  2         .SCB[PSM$L_ENTRY_NUMBER],                     ! entry number
; 2841         P 3764  2         SCB[PSM$Q_QUEUE],                             ! batch que present tense
; 2842         P 3765  2         DATE_QUEUED[0],                               ! time queued
; 2843         P 3766  2         USERNAME_DESC[0],                             ! user name
; 2844         P 3767  2         .SCB[PSM$L_UIC],                              ! user uic
; 2845         P 3768  2         ACCOUNT_DESC[0],                              ! user account
; 2846         P 3769  2         .SCB[PSM$L_PRIORITY],                         ! queue priority
; 2847         P 3770  2         .CHOICE,                                      ! started/completed/restarted
; 2848         P 3771  2         SCB[PSM$Q_DEVICE_NAME],                       ! device name
; 2849         P 3772  2         DATE_PRINTED[0],                              ! time printed
; 2850         P 3773  2         SCB[PSM$Q_EXECUTOR_QUEUE]                     ! executor queue
; 2851           3774  2     );
; 2852           3775  2
; 2853           3776  2 RETURN SS$_NORMAL;
; 2854           3777  1 END;
```

```
                             45  44  4F  4E  24  53  59  53  01317 P.AAY:  .ASCII   \SYS$NODE\                              ;
                                                             0131F         .BLKB    1                                       ;
                                                   00000008  01320 P.AAX:  .LONG    8                                       ;
                                                   00000000' 01324         .ADDRESS P.AAY                                   ;
                                         44  25  37  31  21  01328 P.ABA:  .ASCII   \!17%D\                                 ;
                                                             0132D         .BLKB    3                                       ;
                                                   00000005  01330 P.AAZ:  .LONG    5                                       ;
                                                   00000000' 01334         .ADDRESS P.ABA                                   ;
                             20  53  41  21  20  62  6F  4A  01338 P.ABC:  .ASCII   \Job !AS \                               ;
                                     20  29  4C  55  21  28  01340         .ASCII   \(!UL) \                                 ;
     20  53  41  21  20  6F  74  20  64  65  75  65  75  71  01346         .ASCII   \queued to !AS \                         ;
                                 20  53  41  21  20  6E  6F  01354         .ASCII   \on !AS \                                ;
         20  2C  53  41  21  20  72  65  73  75  20  79  62  0135B         .ASCII   \by user !AS, \                          ;
                         20  2C  49  25  21  20  43  49  55  01368         .ASCII   \UIC !%I, \                              ;
 21  20  74  6E  75  6F  63  63  61  20  72  65  64  6E  75  01371         .ASCII   \under account !AS \                     ;
                                                 20  53  41  01380                                                           ;
 4C  55  21  20  79  74  69  72  6F  69  72  70  20  74  61  01383         .ASCII   \at priority !UL, \                      ;
                                                     20  2C  01392                                                           ;
                                             20  43  41  21  01394         .ASCII   \!AC \                                   ;
 20  53  41  21  20  72  65  74  6E  69  72  70  20  6E  6F  01398         .ASCII   \on printer !AS \                        ;
                                 20  53  41  21  20  6E  6F  013A7         .ASCII   \on !AS \                                ;
     53  41  21  20  65  75  65  75  71  20  6D  6F  72  66  013AE         .ASCII   \from queue !AS\                         ;
                                                         2E  013BC         .ASCII   \.\                                      ;
                                                             013BD         .BLKB    3                                        ;
                                                   00000085  013C0 P.ABB:  .LONG    133                                      ;
                                                   00000000' 013C4         .ADDRESS P.ABC                                    ;
                             64  65  74  72  61  74  73  07  013C8 P.ABD:  .ASCII   <7>\started\                             ;
                     64  65  74  72  61  74  73  65  72  09  013D0 P.ABE:  .ASCII   <9>\restarted\                           ;
                     64  65  74  65  6C  70  6D  6F  63  09  013DA P.ABF:  .ASCII   <9>\completed\                           ;
                             44  45  54  52  4F  42  41  07  013E4 P.ABG:  .ASCII   <7>\ABORTED\                             ;

                                                                   TRAILING=          1
                                                                   LEADING=           0
                                                                   NODE=              P.AAX
                                                                   DATE_FORMAT=       P.AAZ
                                                                   SENT_FORMAT1=      P.ABB


                                                        003C 00000 GET_JOB_DESCRIPTION:
                                                                           .WORD    Save R2,R3,R4,R5                         ; 3630
                                      55      0000V  CF  9E  00002         MOVAB    DISCARD, R5                              ;
                                      54  00000000G  00  9E  00007         MOVAB    SYS$FAO, R4                              ;
                                      5E         B4  AE  9E  0000E         MOVAB    -76(SP), SP                              ;
                                      52         04  AC  D0  00012         MOVL     SCB, R2                                  ; 3677
                                      50       016C  C2  9E  00016         MOVAB    364(R2), R0                              ;
                                  04  AE             60  3C  0001B         MOVZWL   (R0), USERNAME_DESC                      ;
                                  08  AE         04  A0  D0  0001F         MOVL     4(R0), USERNAME_DESC+4                   ; 3678
                                                 08  AE  9F  00024         PUSHAB   USERNAME_DESC+4                          ; 3682
                                                 08  AE  9F  00027         PUSHAB   USERNAME_DESC                            ;
                                                 0C  AE  DD  0002A         PUSHL    USERNAME_DESC                            ; 3681
                                                 14  AE  DD  0002D         PUSHL    USERNAME_DESC+4                          ;
                                                     20  DD  00030         PUSHL    #32                                      ;
                                                     01  DD  00032         PUSHL    #1                                       ;
                                      65             06  FB  00034         CALLS    #6, DISCARD                              ;
                                      50         1C  A2  9E  00037         MOVAB    28(R2), R0                               ; 3686
```

```
            0C  AE        60 3C 0003B      MOVZWL  (R0), ACCOUNT_DESC                  ;
            10  AE    04  A0 D0 0003F      MOVL    4(R0), ACCOUNT_DESC+4               ; 3687
                      10  AE 9F 00044      PUSHAB  ACCOUNT_DESC+4                      ; 3691
                      10  AE 9F 00047      PUSHAB  ACCOUNT_DESC                        ;
                      14  AE DD 0004A      PUSHL   ACCOUNT_DESC                        ; 3690
                      1C  AE DD 0004D      PUSHL   ACCOUNT_DESC+4                      ;
                          20 DD 00050      PUSHL   #32                                 ;
                          01 DD 00052      PUSHL   #1                                  ;
                65        06 FB 00054      CALLS   #6, DISCARD                         ;
                12    08  AC E9 00057      BLBC    TIME_FLAG, 1$                       ; 3694
                53  FF7D  CF 9E 0005B      MOVAB   P.ABD, CHOICE                       ; 3697
  17  0140      C2        02 E1 00060      BBC     #2, 320(R2), 2$                     ; 3698
                53  FF7A  CF 9E 00066      MOVAB   P.ABE, CHOICE                       ; 3700
                          10 11 0006B      BRB     2$                                  ; 3694
                53  FF7D  CF 9E 0006D 1$:  MOVAB   P.ABF, CHOICE                       ; 3708
                    028E  C2 D5 00072      TSTL    654(R2)                             ; 3712
                          05 13 00076      BEQL    2$                                  ;
                53  FF7C  CF 9E 00078      MOVAB   P.ABG, CHOICE                       ; 3717
            44  AE        11 D0 0007D 2$:  MOVL    #17, DATE_QUEUED                    ; 3732
            48  AE    30  AE 9E 00081      MOVAB   DATE_QUEUED_BUFF, DATE_QUEUED+4     ; 3733
                    015C  C2 9F 00086      PUSHAB  348(R2)                             ; 3738
                      48  AE 9F 0008A      PUSHAB  DATE_QUEUED                         ;
                      08  AE 9F 0008D      PUSHAB  RET_LENGTH                          ;
                    FEB0  CF 9F 00090      PUSHAB  DATE_FORMAT                         ;
                64        04 FB 00094      CALLS   #4, SYS$FAO                         ;
                      48  AE 9F 00097      PUSHAB  DATE_QUEUED+4                       ; 3741
                      48  AE 9F 0009A      PUSHAB  DATE_QUEUED                         ;
                      08  AE DD 0009D      PUSHL   RET_LENGTH                          ; 3740
                      54  AE DD 000A0      PUSHL   DATE_QUEUED+4                       ;
                          20 DD 000A3      PUSHL   #32                                 ;
                          7E D4 000A5      CLRL    -(SP)                               ;
                65        06 FB 000A7      CALLS   #6, DISCARD                         ;
                      48  AE 9F 000AA      PUSHAB  DATE_QUEUED+4                       ; 3743
                      48  AE 9F 000AD      PUSHAB  DATE_QUEUED                         ;
                      08  AE DD 000B0      PUSHL   RET_LENGTH                          ; 3742
                      54  AE DD 000B3      PUSHL   DATE_QUEUED+4                       ;
                          20 DD 000B6      PUSHL   #32                                 ;
                          01 DD 000B8      PUSHL   #1                                  ;
                65        06 FB 000BA      CALLS   #6, DISCARD                         ;
            28  AE        11 D0 000BD      MOVL    #17, DATE_PRINTED                   ; 3746
            2C  AE    14  AE 9E 000C1      MOVAB   DATE_PRINTED_BUFF, DATE_PRINTED+4   ; 3747
                    0234  C2 9F 000C6      PUSHAB  564(R2)                             ; 3752
                      2C  AE 9F 000CA      PUSHAB  DATE_PRINTED                        ;
                      08  AE 9F 000CD      PUSHAB  RET_LENGTH                          ;
                    FE70  CF 9F 000D0      PUSHAB  DATE_FORMAT                         ;
                64        04 FB 000D4      CALLS   #4, SYS$FAO                         ;
                      2C  AE 9F 000D7      PUSHAB  DATE_PRINTED+4                      ; 3755
                      2C  AE 9F 000DA      PUSHAB  DATE_PRINTED                        ;
                      08  AE DD 000DD      PUSHL   RET_LENGTH                          ; 3754
                      38  AE DD 000E0      PUSHL   DATE_PRINTED+4                      ;
                          20 DD 000E3      PUSHL   #32                                 ;
                          7E D4 000E5      CLRL    -(SP)                               ;
                65        06 FB 000E7      CALLS   #6, DISCARD                         ;
                      2C  AE 9F 000EA      PUSHAB  DATE_PRINTED+4                      ; 3757
                      2C  AE 9F 000ED      PUSHAB  DATE_PRINTED                        ;
                      08  AE DD 000F0      PUSHL   RET_LENGTH                          ; 3756
                      38  AE DD 000F3      PUSHL   DATE_PRINTED+4                      ;
```

```
                                 20  DD 000F6           PUSHL    #32                          ;
                                 01  DD 000F8           PUSHL    #1                           ;
                    65           06  FB 000FA           CALLS    #6, DISCARD                  ;
                          5C     A2  9F 000FD           PUSHAB   92(R2)                       ; 3774
                          2C     AE  9F 00100           PUSHAB   DATE_PRINTED                 ;
                          4C     A2  9F 00103           PUSHAB   76(R2)                       ;
                                 53  DD 00106           PUSHL    CHOICE                       ;
                        0128     C2  DD 00108           PUSHL    296(R2)                      ;
                          20     AE  9F 0010C           PUSHAB   ACCOUNT_DESC                 ;
                        0168     C2  DD 0010F           PUSHL    360(R2)                      ;
                          20     AE  9F 00113           PUSHAB   USERNAME_DESC                ;
                          64     AE  9F 00116           PUSHAB   DATE_QUEUED                  ;
                        012C     C2  9F 00119           PUSHAB   300(R2)                      ;
                          58     A2  DD 0011D           PUSHL    88(R2)                       ;
                        00A8     C2  9F 00120           PUSHAB   168(R2)                      ;
                          0C     AC  DD 00124           PUSHL    STR_DESC                     ;
                          10     AC  DD 00127           PUSHL    RET_LEN                      ;
                        FEA6     CF  9F 0012A           PUSHAB   SENT_FORMAT1                 ;
                    64           0F  FB 0012E           CALLS    #15, SYS$FAO                 ;
                                     04 00131           RET                                   ; 3777

; Routine Size:  306 bytes,    Routine Base:  CODE + 13EC
```

```
  1 %sbttl 'GET_FILE_DESCRIPTION - Create a Sentence Describing the Current File'
  1 !++
  1 ! Functional Description:
  1 !       This routine creates a sentence describing the current File.
  1 !
  1 ! Formal Parameters:
  1 !       SCB             - Address of the SCB
  1 !       STR_DESC        - Desc of String to Return
  1 !       RET_LEN         - Return length of Desc.
  1 !
  1 ! Implicit Inputs:
  1 !               none
  1 !
  1 ! Implicit Outputs:
  1 !               none
  1 !
  1 ! Returned Value:
  1 !               none
  1 !
  1 ! Side Effects:
  1 !               none
  1 !--
  1 ROUTINE GET_FILE_DESCRIPTION    (
  1                 SCB             : REF $BBLOCK,          ! SCB
  1                 STR_DESC        : REF VECTOR[2],        ! Output buffer desc
  1                 RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
  1                         )       : NOVALUE =
  2 BEGIN
  2 BIND
  2     FAB = .SCB[PSM$A_FAB]: $BBLOCK,
  2     NAM = .SCB[PSM$A_NAM]: $BBLOCK,
  2     XABDAT = .SCB[PSM$A_XABDAT]: $BBLOCK,
  2     XABFHC = .SCB[PSM$A_XABFHC]: $BBLOCK,
  2     XABPRO = .SCB[PSM$A_XABPRO]: $BBLOCK,
  2
P 2     FORMAT_POS = $DESCRIPTOR (
P 2         'File !AS ',                                ! - file name
P 2         '(!UL,!UL,!UL), ',                          ! - file Id number
P 2         'last revised on !17%D, ',                  ! - revision date
P 2         'is a !UL block ',                          ! - file size
P 2         '!AC file ',                                ! - file organization
  2         'owned by UIC !%I.  '),                     ! - owner user uic
  2
P 2     RECORD_FORMAT = $DESCRIPTOR (
P 2         'The records are ',     ! -
  2         '!AC with '),                               ! - record format
  2
P 2     RECORD_VFC_FORMAT = $DESCRIPTOR (
P 2         'The records are ',     ! -
P 2         'variable length with a ',
  2         'fixed control size of !UL byte!%S and '),
  2                                                     ! - fixed control area size
  2
P 2     REC_SIZE = $DESCRIPTOR (
  2         '   The longest record is !UL byte!%S.'),
  2                                                     ! - max record size
  2
```

```
; 2913    P 3835  2       CARRIAGE_FORMAT = $DESCRIPTOR (
; 2914      3836  2           '!AC.'),                                    ! - record attributes
; 2915      3837  2
; 2916    P 3838  2       FORMAT_NEG = $DESCRIPTOR (
; 2917      3839  2           'File (!AS) description is unavailable to the symbiont.');
; 2918      3840  2
; 2919      3841  2   LITERAL
; 2920      3842  2       K_MAX_BUFFER_SIZE = 512;
; 2921      3843  2
; 2922      3844  2   LOCAL
; 2923      3845  2       RECORD_SIZE         ,
; 2924      3846  2       FILE_SIZE           ,
; 2925      3847  2       ORGANIZATION        ,
; 2926      3848  2       ATTRIBUTES          ,
; 2927      3849  2       FORMAT              ,
; 2928      3850  2       CURRENT_LEN         : INITIAL (0),
; 2929      3851  2       DATE_REVISED        : VECTOR[2],
; 2930      3852  2       STRING_PTR          : VECTOR [2];           ! Pointer to current string
; 2931      3853  2
; 2932      3854  2   ! Allocate the buffer for "GET_xxx" Routines
; 2933      3855  2   !
; 2934      3856  2   STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE;
; 2935      3857  2   STRING_PTR[ADDR] = .STR_DESC[ADDR];     ! init address
; 2936      3858  2
; 2937      3859  2   RET_LEN[0] = 0;
; 2938      3860  2
; 2939      3861  2   IF FILE_OPEN(.SCB)
; 2940      3862  2   THEN
; 2941      3863  3       BEGIN
; 2942      3864  3       ! get the file size
; 2943      3865  3       !
; 2944      3866  3       FILE_SIZE = .XABFHC[XAB$L_EBK];
; 2945      3867  3       IF (.XABFHC[XAB$W_FFB] EQL 0) AND
; 2946      3868  4          ( .FILE_SIZE NEQ 0)
; 2947      3869  3       THEN
; 2948      3870  3           FILE_SIZE = .FILE_SIZE - 1;
; 2949      3871  3
; 2950      3872  3       ! insert file organization
; 2951      3873  3       !
; 2952      3874  3       IF .FAB[FAB$B_ORG] EQL FAB$C_IDX
; 2953      3875  3       THEN
; 2954      3876  3           ORGANIZATION = UPLIT BYTE (%ASCIC 'indexed')
; 2955      3877  3       ELSE IF .FAB[FAB$B_ORG] EQL FAB$C_SEQ
; 2956      3878  3       THEN
; 2957      3879  3           ORGANIZATION = UPLIT BYTE (%ASCIC 'sequential')
; 2958      3880  3       ELSE IF .FAB[FAB$B_ORG] EQL FAB$C_REL
; 2959      3881  3       THEN
; 2960      3882  3           ORGANIZATION = UPLIT BYTE (%ASCIC 'relative')
; 2961      3883  3       ELSE
; 2962      3884  3           ORGANIZATION = UPLIT BYTE (%ASCIC 'undefined organization');
; 2963      3885  3
; 2964      3886  3
; 2965    P 3887  3       $FAO (  FORMAT_POS,
; 2966    P 3888  3               CURRENT_LEN,
; 2967    P 3889  3               STRING_PTR[0],
; 2968    P 3890  3               SCB[PSM$Q_FILE_SPECIFICATION],
; 2969    P 3891  3               .NAM[NAM$Q_FID_NUM],
; 2970      P 3892  3               .NAM[NAM$W_FID_SEQ],
; 2971      P 3893  3               .NAM[NAM$W_FID_RVN],
; 2972      P 3894  3               XABDAT[XAB$Q_RDT],
; 2973      P 3895  3               .FILE_SIZE,
; 2974      P 3896  3               .ORGANIZATION,
; 2975      P 3897  3               .XABPRO[XAB$L_UIC]
; 2976        3898  3           );
; 2977        3899  3
; 2978        3900  3       RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 2979        3901  3       STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 2980        3902  3       STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 2981        3903  3
; 2982        3904  3       ! get record format
; 2983        3905  3       !
; 2984        3906  3
; 2985        3907  3       RECORD_SIZE = .XABFHC[XAB$W_LRL];             ! record size
; 2986        3908  3
; 2987        3909  3       IF .FAB[FAB$B_RFM] NEQ FAB$C_VFC
; 2988        3910  3       THEN
; 2989        3911  4           BEGIN
; 2990        3912  4           ! get record type
; 2991        3913  4           !
; 2992        3914  4           SELECTONE .FAB[FAB$B_RFM] OF
; 2993        3915  4               SET
; 2994        3916  4               [FAB$C_FIX]:            FORMAT = UPLIT BYTE
; 2995        3917  4                                                (%ASCIC 'fixed-length');
; 2996        3918  4               [FAB$C_STM]:            FORMAT = UPLIT BYTE
; 2997        3919  4                                                (%ASCIC 'stream');
; 2998        3920  4               [FAB$C_STMCR]:          FORMAT = UPLIT BYTE
; 2999        3921  4                                                (%ASCIC 'stream-CR');
; 3000        3922  4               [FAB$C_STMLF]:          FORMAT = UPLIT BYTE
; 3001        3923  4                                                (%ASCIC 'stream-LF');
; 3002        3924  4               [FAB$C_UDF]:            FORMAT = UPLIT BYTE
; 3003        3925  4                                                (%ASCIC 'an undefined format');
; 3004        3926  4               [FAB$C_VAR]:            FORMAT = UPLIT BYTE
; 3005        3927  4                                                (%ASCIC 'variable length');
; 3006        3928  4               TES;
; 3007        3929  4
; 3008      P 3930  4           $FAO (  RECORD_FORMAT,
; 3009      P 3931  4                   CURRENT_LEN,
; 3010      P 3932  4                   STRING_PTR[0],
; 3011      P 3933  4                   .FORMAT
; 3012        3934  4               );
; 3013        3935  4
; 3014        3936  4           END
; 3015        3937  3       ELSE
; 3016      P 3938  3           $FAO (  RECORD_VFC_FORMAT,
; 3017      P 3939  3                   CURRENT_LEN,
; 3018      P 3940  3                   STRING_PTR[0],
; 3019      P 3941  3                   .FAB[FAB$B_FSZ]
; 3020        3942  3               );
; 3021        3943  3
; 3022        3944  3       RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 3023        3945  3       STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 3024        3946  3       STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 3025        3947  3
; 3026        3948  3       ! get carriage control
```

```
  3         !
  3         IF .FAB[FAB$V_CR]
  3         THEN
  3             ATTRIBUTES = UPLIT BYTE (%ASCIC 'implied (CR) carriage control')
  3         ELSE
  4             BEGIN
  4             IF .FAB[FAB$V_FTN]
  4             THEN
  4                 ATTRIBUTES = UPLIT BYTE (%ASCIC 'FORTRAN (FTN) carriage control')
  4             ELSE
  5                 BEGIN
  5                 IF .FAB[FAB$V_PRN]
  5                 THEN
  5                     ATTRIBUTES =
  5                             UPLIT BYTE (%ASCIC 'print file (PRN) carriage control')
  5                 ELSE
  5                     ATTRIBUTES = UPLIT BYTE
  5                             (%ASCIC 'imbedded (<none>) carriage control');
  4                 END;
  3             END;
  3
P 3         $FAO (
P 3             CARRIAGE_FORMAT,
P 3             CURRENT_LEN,
P 3             STRING_PTR[0],
  3             .ATTRIBUTES );
  3
  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
  3
  3         IF .FAB[FAB$B_RFM] NEQ FAB$C_FIX
  3         THEN
  4             BEGIN
P 4             $FAO (
P 4                 REC_SIZE,
P 4                 CURRENT_LEN,
P 4                 STRING_PTR[0],
  4                 .RECORD_SIZE );
  4
  4             RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
  4             STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
  4             STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
  3             END;
  3
  3         END
  2     ELSE
  3         BEGIN
P 3         $FAO (
P 3             FORMAT_NEG,
P 3             CURRENT_LEN,
P 3             STRING_PTR[0],
P 3             SCB[PSM$Q_FILE_SPECIFICATION]
  3             );
  3
  3         RET_LEN[0] = .CURRENT_LEN;
  2         END;
```

```
; 3084          4006  2
; 3085          4007  2 ! final check for overflow
; 3086          4008  2 IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 3087          4009  2 THEN
; 3088          4010  3     BEGIN
; 3089          4011  3         RET_LEN[0] = 512;
; 3090          4012  3         RETURN;
; 3091          4013  2     END;
; 3092          4014  2
; 3093          4015  2 RETURN SS$_NORMAL;
; 3094          4016  1 END;



                        20  53  41  21  20  65  6C  69  46  0151E P.ABI:   .ASCII  \File !AS \
20  2C  29  4C  55  21  2C  4C  55  21  2C  4C  55  21  28  01527          .ASCII  \(!UL,!UL,!UL), \                         ;
6E  6F  20  64  65  73  69  76  65  72  20  74  73  61  6C  01536          .ASCII  \last revised on !17%D, \                 ;
                            20  2C  44  25  37  31  21  20  01545                                                            ;
20  6B  63  6F  6C  62  20  4C  55  21  20  61  20  73  69  0154D          .ASCII  \is a !UL block \                         ;
                            20  65  6C  69  66  20  43  41  21  0155C      .ASCII  \!AC file \                               ;
25  21  20  43  49  55  20  79  62  20  64  65  6E  77  6F  01565          .ASCII  \owned by UIC !%I.  \                     ;
                                            20  20  2E  49  01574                                                            ;
                                            0000005A  01578 P.ABH:   .LONG   90                                       ;
                                            00000000' 0157C          .ADDRESS P.ABI                                   ;
65  72  61  20  73  64  72  6F  63  65  72  20  65  68  54  01580 P.ABK:   .ASCII  \The records are \                  ;
                                                        20  0158F                                                            ;
                        20  68  74  69  77  20  43  41  21  01590          .ASCII  \!AC with \                               ;
                                                            01599          .BLKB   3
                                            00000019  0159C P.ABJ:   .LONG   25                                       ;
                                            00000000' 015A0          .ADDRESS P.ABK                                   ;
65  72  61  20  73  64  72  6F  63  65  72  20  65  68  54  015A4 P.ABM:   .ASCII  \The records are \                  ;
                                                        20  015B3                                                            ;
68  74  67  6E  65  6C  20  65  6C  62  61  69  72  61  76  015B4          .ASCII  \variable length with a \                 ;
                            20  61  20  68  74  69  77  20  015C3                                                            ;
73  20  6C  6F  72  74  6E  6F  63  20  64  65  78  69  66  015CB          .ASCII  \fixed control size of !UL byte!%S and \  ;
65  74  79  62  20  4C  55  21  20  66  6F  20  65  7A  69  015DA                                                            ;
                            20  64  6E  61  20  53  25  21  015E9                                                            ;
                                                            015F1          .BLKB   3
                                            0000004D  015F4 P.ABL:   .LONG   77                                       ;
                                            00000000' 015F8          .ADDRESS P.ABM                                   ;
72  20  74  73  65  67  6E  6F  6C  20  65  68  54  20  20  015FC P.ABO:   .ASCII  \  The longest record is !UL byte!%S.\ ;
79  62  20  4C  55  21  20  73  69  20  64  72  6F  63  65  0160B                                                            ;
                                    2E  53  25  21  65  74  0161A                                                            ;
                                            00000024  01620 P.ABN:   .LONG   36                                       ;
                                            00000000' 01624          .ADDRESS P.ABO                                   ;
                                            2E  43  41  21  01628 P.ABQ:   .ASCII  \!AC.\                             ;
                                            00000004  0162C P.ABP:   .LONG   4                                        ;
                                            00000000' 01630          .ADDRESS P.ABQ                                   ;
63  73  65  64  20  29  53  41  21  28  20  65  6C  69  46  01634 P.ABS:   .ASCII  \File (!AS) description is unavailable to\ ;
76  61  6E  75  20  73  69  20  6E  6F  69  74  70  69  72  01643                                                            ;
                        6F  74  20  65  6C  62  61  6C  69  61  01652                                                        ;
    2E  74  6E  6F  69  62  6D  79  73  20  65  68  74  20  0165C          .ASCII  \ the symbiont.\                          ;
                                                            0166A          .BLKB   2
                                            00000036  0166C P.ABR:   .LONG   54                                       ;
                                            00000000' 01670          .ADDRESS P.ABS                                   ;
                        64  65  78  65  64  6E  69  07  01674 P.ABT:   .ASCII  <7>\indexed\                                 ;
            6C  61  69  74  6E  65  75  71  65  73  0A  0167C P.ABU:   .ASCII  <10>\sequential\                             ;
```

```
                        65  76  69  74  61  6C  65  72  08 01687 P.ABV:   .ASCII  <8>\relative\                               ;
61  67  72  6F  20  64  65  6E  69  66  65  64  6E  75  16 01690 P.ABW:   .ASCII  <22>\undefined organization\                ;
                            6E  6F  69  74  61  7A  69  6E 0169F                                                              ;
        68  74  67  6E  65  6C  2D  64  65  78  69  66  0C 016A7 P.ABX:   .ASCII  <12>\fixed-length\                          ;
                                6D  61  65  72  74  73  06 016B4 P.ABY:   .ASCII  <6>\stream\                                 ;
                    52  43  2D  6D  61  65  72  74  73  09 016BB P.ABZ:   .ASCII  <9>\stream-CR\                              ;
                    46  4C  2D  6D  61  65  72  74  73  09 016C5 P.ACA:   .ASCII  <9>\stream-LF\                              ;
66  20  64  65  6E  69  66  65  64  6E  75  20  6E  61  13 016CF P.ACB:   .ASCII  <19>\an undefined format\                   ;
                                        74  61  6D  72  6F 016DE                                                              ;
74  67  6E  65  6C  20  65  6C  62  61  69  72  61  76  0F 016E3 P.ACC:   .ASCII  <15>\variable length\                       ;
                                                        68 016F2                                                              ;
63  20  29  52  43  28  20  64  65  69  6C  70  6D  69  1D 016F3 P.ACD:   .ASCII  <29>\implied (CR) carriage control\         ;
6C  6F  72  74  6E  6F  63  20  65  67  61  69  72  72  61 01702                                                              ;
20  29  4E  54  46  28  20  4E  41  52  54  52  4F  46  1E 01711 P.ACE:   .ASCII  <30>\FORTRAN (FTN) carriage control\        ;
6F  72  74  6E  6F  63  20  65  67  61  69  72  72  61  63 01720                                                              ;
                                                        6C 0172F                                                              ;
52  50  28  20  65  6C  69  66  20  74  6E  69  72  70  21 01730 P.ACF:   .ASCII  \!print file (PRN) carriage control\        ;
6E  6F  63  20  65  67  61  69  72  72  61  63  20  29  4E 0173F                                                              ;
                                            6C  6F  72  74 0174E                                                              ;
6E  6F  6E  3C  28  20  64  65  64  64  65  62  6D  69  22 01752 P.ACG:   .ASCII  \"imbedded (<none>) carriage control\       ;
6F  63  20  65  67  61  69  72  72  61  63  20  29  3E  65 01761                                                              ;
                                        6C  6F  72  74  6E 01770                                                              ;

                                                                 FORMAT_POS=         P.ABH
                                                                 RECORD_FORMAT=      P.ABJ
                                                                 RECORD_VFC_FORMAT=  P.ABL
                                                                 REC_SIZE=           P.ABN
                                                                 CARRIAGE_FORMAT=    P.ABP
                                                                 FORMAT_NEG=         P.ABR


                                                      07FC 00000 GET_FILE_DESCRIPTION:
                                                                          .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10            ; 3800
                                     5A  00000000G   00 9E 00002          MOVAB   SYS$FAO, R10                                ;
                                     59       FEF2   CF 9E 00009          MOVAB   P.ABT, R9                                   ;
                                     5E              10 C2 0000E          SUBL2   #16, SP                                     ;
                                     50         04   AC 7D 00011          MOVQ    SCB, R0                                     ; 3807
                                     52       0248   C0 D0 00015          MOVL    584(R0), R2                                 ;
                                     54       024C   C0 D0 0001A          MOVL    588(R0), R4                                 ; 3808
                                     57       0254   C0 D0 0001F          MOVL    596(R0), R7                                 ; 3809
                                     53       0258   C0 D0 00024          MOVL    600(R0), R3                                 ; 3810
                                     56       025C   C0 D0 00029          MOVL    604(R0), R6                                 ; 3811
                                                     7E D4 0002E          CLRL    CURRENT_LEN                                 ;
                                 04  AE       0200   8F 3C 00030          MOVZWL  #512, STRING_PTR                            ; 3856
                                 08  AE         04   A1 D0 00036          MOVL    4(R1), STRING_PTR+4                         ; 3857
                                     55         0C   AC D0 0003B          MOVL    RET_LEN, R5                                 ; 3859
                                                     65 B4 0003F          CLRW    (R5)                                        ;
                                     58       0098   C0 9E 00041          MOVAB   152(R0), R8                                 ; 3898
                                                     50 DD 00046          PUSHL   R0                                          ; 3861
                              0000V  CF              01 FB 00048          CALLS   #1, FILE_OPEN                               ;
                                     03              50 E8 0004D          BLBS    R0, 1$                                      ;
                                                   0162 31 00050          BRW     19$                                         ;
                                     51         10   A3 D0 00053 1$:      MOVL    16(R3), FILE_SIZE                           ; 3866
                                                14   A3 B5 00057          TSTW    20(R3)                                      ; 3867
                                                     06 12 0005A          BNEQ    2$                                          ;
                                                     51 D5 0005C          TSTL    FILE_SIZE                                   ; 3868
                                                     02 13 0005E          BEQL    2$                                          ;
```

```
                                    51  D7 00060            DECL    FILE_SIZE                                  ; 3870
                        20      1D  A2  91 00062 2$:        CMPB    29(R2), #32                                ; 3874
                                    05  12 00066            BNEQ    3$
                        50          69  9E 00068            MOVAB   P.ABT, ORGANIZATION                        ; 3876
                                    1B  11 0006B            BRB     6$
                                1D  A2  95 0006D 3$:        TSTB    29(R2)                                     ; 3877
                                    06  12 00070            BNEQ    4$
                        50      08  A9  9E 00072            MOVAB   P.ABU, ORGANIZATION                        ; 3879
                                    10  11 00076            BRB     6$
                        10      1D  A2  91 00078 4$:        CMPB    29(R2), #16                                ; 3880
                                    06  12 0007C            BNEQ    5$
                        50      13  A9  9E 0007E            MOVAB   P.ABV, ORGANIZATION                        ; 3882
                                    04  11 00082            BRB     6$
                        50      1C  A9  9E 00084 5$:        MOVAB   P.ABW, ORGANIZATION                        ; 3884
                                0C  A6  DD 00088 6$:        PUSHL   12(R6)                                     ; 3898
                                    50  DD 0008B            PUSHL   ORGANIZATION
                                    51  DD 0008D            PUSHL   FILE_SIZE
                                0C  A7  9F 0008F            PUSHAB  12(R7)
                        7E      28  A4  3C 00092            MOVZWL  40(R4), -(SP)
                        7E      26  A4  3C 00096            MOVZWL  38(R4), -(SP)
                        7E      24  A4  3C 0009A            MOVZWL  36(R4), -(SP)
                                    58  DD 0009E            PUSHL   R8
                                24  AE  9F 000A0            PUSHAB  STRING_PTR
                                24  AE  9F 000A3            PUSHAB  CURRENT_LEN
                              FF04  C9  9F 000A6            PUSHAB  FORMAT_POS
                        6A          0B  FB 000AA            CALLS   #11, SYS$FAO
                        65          6E  A0 000AD            ADDW2   CURRENT_LEN, (R5)                          ; 3900
                    08  AE          6E  C0 000B0            ADDL2   CURRENT_LEN, STRING_PTR+4                  ; 3901
                    04  AE          65  3C 000B4            MOVZWL  (R5), STRING_PTR                           ; 3902
    04    AE 00000200   8F      04  AE  C3 000B8            SUBL3   STRING_PTR, #512, STRING_PTR
                        54      0A  A3  3C 000C2            MOVZWL  10(R3), RECORD_SIZE                        ; 3907
                        53      1F  A2  9A 000C6            MOVZBL  31(R2), R3                                 ; 3909
                        03          53  91 000CA            CMPB    R3, #3
                                    4D  13 000CD            BEQL    13$
                        01          53  91 000CF            CMPB    R3, #1                                     ; 3916
                                    06  12 000D2            BNEQ    7$
                        50      33  A9  9E 000D4            MOVAB   P.ABX, FORMAT
                                    34  11 000D8            BRB     12$
                        04          53  91 000DA 7$:        CMPB    R3, #4                                     ; 3918
                                    06  12 000DD            BNEQ    8$
                        50      40  A9  9E 000DF            MOVAB   P.ABY, FORMAT
                                    29  11 000E3            BRB     12$
                        06          53  91 000E5 8$:        CMPB    R3, #6                                     ; 3920
                                    06  12 000E8            BNEQ    9$
                        50      47  A9  9E 000EA            MOVAB   P.ABZ, FORMAT
                                    1E  11 000EE            BRB     12$
                        05          53  91 000F0 9$:        CMPB    R3, #5                                     ; 3922
                                    06  12 000F3            BNEQ    10$
                        50      51  A9  9E 000F5            MOVAB   P.ACA, FORMAT
                                    13  11 000F9            BRB     12$
                                    53  D5 000FB 10$:       TSTL    R3                                         ; 3924
                                    06  12 000FD            BNEQ    11$
                        50      5B  A9  9E 000FF            MOVAB   P.ACB, FORMAT
                                    09  11 00103            BRB     12$
                        02          53  91 00105 11$:       CMPB    R3, #2                                     ; 3926
                                    04  12 00108            BNEQ    12$
                        50      6F  A9  9E 0010A            MOVAB   P.ACC, FORMAT
```

```
                                                       50  DD 0010E 12$:     PUSHL   FORMAT                                          ; 3934
                                                  08   AE  9F 00110          PUSHAB  STRING_PTR                                      ;
                                                  08   AE  9F 00113          PUSHAB  CURRENT_LEN                                     ;
                                                FF28   C9  9F 00116          PUSHAB  RECORD_FORMAT                                   ;
                                                       0D  11 0011A          BRB     14$                                             ;
                                        7E        3F   A2  9A 0011C 13$:     MOVZBL  63(R2), -(SP)                                   ; 3942
                                                  08   AE  9F 00120          PUSHAB  STRING_PTR                                      ;
                                                  08   AE  9F 00123          PUSHAB  CURRENT_LEN                                     ;
                                                  80   A9  9F 00126          PUSHAB  RECORD_VFC_FORMAT                               ;
                                        6A             04  FB 00129 14$:     CALLS   #4, SYS$FAO                                     ;
                                        65             6E  A0 0012C          ADDW2   CURRENT_LEN, (R5)                               ; 3944
                                   08   AE             6E  C0 0012F          ADDL2   CURRENT_LEN, STRING_PTR+4                       ; 3945
                                   04   AE             65  3C 00133          MOVZWL  (R5), STRING_PTR                                ; 3946
                     04   AE 00000200   8F        04   AE  C3 00137          SUBL3   STRING_PTR, #512, STRING_PTR                    ;
                          06       1E   A2             01  E1 00141          BBC     #1, 30(R2), 15$                                 ; 3950
                                        50        7F   A9  9E 00146          MOVAB   P.ACD, ATTRIBUTES                               ; 3952
                                                       1C  11 0014A          BRB     18$                                             ;
                                        07        1E   A2  E9 0014C 15$:     BLBC    30(R2), 16$                                     ; 3955
                                        50      009D   C9  9E 00150          MOVAB   P.ACE, ATTRIBUTES                               ; 3957
                                                       11  11 00155          BRB     18$                                             ;
                          07       1E   A2             02  E1 00157 16$:     BBC     #2, 30(R2), 17$                                 ; 3960
                                        50      00BC   C9  9E 0015C          MOVAB   P.ACF, ATTRIBUTES                               ; 3963
                                                       05  11 00161          BRB     18$                                             ; 3962
                                        50      00DE   C9  9E 00163 17$:     MOVAB   P.ACG, ATTRIBUTES                               ; 3965
                                                       50  DD 00168 18$:     PUSHL   ATTRIBUTES                                      ; 3974
                                                  08   AE  9F 0016A          PUSHAB  STRING_PTR                                      ;
                                                  08   AE  9F 0016D          PUSHAB  CURRENT_LEN                                     ;
                                                  B8   A9  9F 00170          PUSHAB  CARRIAGE_FORMAT                                 ;
                                        6A             04  FB 00173          CALLS   #4, SYS$FAO                                     ;
                                        65             6E  A0 00176          ADDW2   CURRENT_LEN, (R5)                               ; 3976
                                   08   AE             6E  C0 00179          ADDL2   CURRENT_LEN, STRING_PTR+4                       ; 3977
                                   04   AE             65  3C 0017D          MOVZWL  (R5), STRING_PTR                                ; 3978
                     04   AE 00000200   8F        04   AE  C3 00181          SUBL3   STRING_PTR, #512, STRING_PTR                    ;
                                        01             53  91 0018B          CMPB    R3, #1                                          ; 3980
                                                       36  13 0018E          BEQL    20$                                             ;
                                                       54  DD 00190          PUSHL   RECORD_SIZE                                     ; 3987
                                                  08   AE  9F 00192          PUSHAB  STRING_PTR                                      ;
                                                  08   AE  9F 00195          PUSHAB  CURRENT_LEN                                     ;
                                                  AC   A9  9F 00198          PUSHAB  REC_SIZE                                        ;
                                        6A             04  FB 0019B          CALLS   #4, SYS$FAO                                     ;
                                        65             6E  A0 0019E          ADDW2   CURRENT_LEN, (R5)                               ; 3989
                                   08   AE             6E  C0 001A1          ADDL2   CURRENT_LEN, STRING_PTR+4                       ; 3990
                                   04   AE             65  3C 001A5          MOVZWL  (R5), STRING_PTR                                ; 3991
                     04   AE 00000200   8F        04   AE  C3 001A9          SUBL3   STRING_PTR, #512, STRING_PTR                    ;
                                                       11  11 001B3          BRB     20$                                             ; 3861
                                                       58  DD 001B5 19$:     PUSHL   R8                                              ; 4002
                                                  08   AE  9F 001B7          PUSHAB  STRING_PTR                                      ;
                                                  08   AE  9F 001BA          PUSHAB  CURRENT_LEN                                     ;
                                                  F8   A9  9F 001BD          PUSHAB  FORMAT_NEG                                      ;
                                        6A             04  FB 001C0          CALLS   #4, SYS$FAO                                     ;
                                        65             6E  B0 001C3          MOVW    CURRENT_LEN, (R5)                               ; 4004
                                 0200   8F             65  B1 001C6 20$:     CMPW    (R5), #512                                      ; 4008
                                                       05  1B 001CB          BLEQU   21$                                             ;
                                        65      0200   8F  B0 001CD          MOVW    #512, (R5)                                      ; 4011
                                                           04 001D2 21$:     RET                                                     ; 4016

; Routine Size:  467 bytes,    Routine Base:  CODE + 1775
```

```
; 3096        4017  1 %sbttl 'GET_FILE_NAME - Get Name of the Current File'
; 3097        4018  1 !++
; 3098        4019  1 ! Functional Description:
; 3099        4020  1 !       This routine creates a phrase with the name of the current file.
; 3100        4021  1 !
; 3101        4022  1 ! Formal Parameters:
; 3102        4023  1 !       SCB             - Address of the SCB
; 3103        4024  1 !       STR_DESC        - Desc of String to Return
; 3104        4025  1 !       RET_LEN         - Return length of Desc.
; 3105        4026  1 !
; 3106        4027  1 ! Implicit Inputs:
; 3107        4028  1 !               none
; 3108        4029  1 !
; 3109        4030  1 ! Implicit Outputs:
; 3110        4031  1 !               none
; 3111        4032  1 !
; 3112        4033  1 ! Returned Value:
; 3113        4034  1 !               none
; 3114        4035  1 !
; 3115        4036  1 ! Side Effects:
; 3116        4037  1 !               none
; 3117        4038  1 !--
; 3118        4039  1 ROUTINE GET_FILE_NAME (
; 3119        4040  1                 SCB             : REF $BBLOCK,          ! SCB
; 3120        4041  1                 EXPECTED_LEN    ,                       ! Maximum length allowed
; 3121        4042  1                 STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 3122        4043  1                 RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 3123        4044  1                         )       : NOVALUE =
; 3124        4045  2 BEGIN
; 3125        4046  2 BIND
; 3126      P 4047  2     SENT_FORMAT = $DESCRIPTOR (
; 3127        4048  2         '!AS');
; 3128        4049  2
; 3129        4050  2 LOCAL
; 3130        4051  2     BUFFER      : VECTOR [512,byte],
; 3131        4052  2     LENG        : VECTOR [1],
; 3132        4053  2     NAME        :  VECTOR[2];
; 3133        4054  2
; 3134        4055  2 NAME[SIZE] = %ALLOCATION(BUFFER);       ! allocate for routines
; 3135        4056  2 NAME[ADDR] = BUFFER;                    ! init address
; 3136        4057  2
; 3137      P 4058  2 $FAO (  SENT_FORMAT,
; 3138      P 4059  2         NAME[SIZE],                     ! return length
; 3139      P 4060  2         NAME,                           ! address of string
; 3140      P 4061  2         SCB[PSM$Q_FILE_SPECIFICATION],  ! file name
; 3141        4062  2     );
; 3142        4063  2
; 3143        4064  2 LENG[0] = .EXPECTED_LEN;        !  must be reference to word for call
; 3144        4065  2
; 3145        4066  2 IF .RET_LEN[0] GTR .EXPECTED_LEN THEN
; 3146        4067  2     !  Trim the file spec to fit.
; 3147        4068  2     !
; 3148        4069  2     LIB$TRIM_FILESPEC ( NAME, STR_DESC[0], LENG[0],
; 3149        4070  2                         STR_DESC[SIZE])
; 3150        4071  2 ELSE
; 3151        4072  3     BEGIN
; 3152        4073  3     STR_DESC[SIZE] = .NAME[SIZE];
; 3153          4074  3     STR_DESC[ADDR] = .NAME[ADDR];
; 3154          4075  2     END;
; 3155          4076  2
; 3156          4077  2 RET_LEN[0] = .STR_DESC[SIZE];
; 3157          4078  2
; 3158          4079  2 RETURN SS$_NORMAL;
; 3159          4080  1 END;



                                                 53  41  21 01948 P.ACI:  .ASCII  \!AS\                                       ;
                                                            0194B         .BLKB   1
                                                   00000003 0194C P.ACH:  .LONG   3                                           ;
                                                  00000000' 01950         .ADDRESS P.ACI                                      ;

                                                                  SENT_FORMAT=        P.ACH


                                                       0004 00000 GET_FILE_NAME:
                                                                          .WORD   Save R2                                     ; 4039
                                       5E      FDF4  CE  9E 00002         MOVAB   -524(SP), SP                                ;
                                   04  AE      0200  8F  3C 00007         MOVZWL  #512, NAME                                  ; 4055
                                   08  AE        0C  AE  9E 0000D         MOVAB   BUFFER, NAME+4                              ; 4056
                         7E        04  AC  00000098  8F  C1 00012         ADDL3   #152, SCB, -(SP)                            ; 4062
                                                 08  AE  9F 0001B         PUSHAB  NAME                                        ;
                                                 0C  AE  9F 0001E         PUSHAB  NAME                                        ;
                                                 D4  AF  9F 00021         PUSHAB  SENT_FORMAT                                 ;
                            00000000G  00            04  FB 00024         CALLS   #4, SYS$FAO                                 ;
                                       6E        08  AC  D0 0002B         MOVL    EXPECTED_LEN, LENG                          ; 4064
                                       52        0C  AC  D0 0002F         MOVL    STR_DESC, R2                                ; 4070
      08   AC       10   BC            10            00  ED 00033         CMPZV   #0, #16, @RET_LEN, EXPECTED_LEN             ; 4066
                                                     13  15 0003A         BLEQ    1$                                          ;
                                                     52  DD 0003C         PUSHL   R2                                          ; 4070
                                                 04  AE  9F 0003E         PUSHAB  LENG                                        ; 4069
                                                     52  DD 00041         PUSHL   R2                                          ; 4070
                                                 10  AE  9F 00043         PUSHAB  NAME                                        ; 4069
                            00000000G  00            04  FB 00046         CALLS   #4, LIB$TRIM_FILESPEC                       ; 4070
                                                     04  11 0004D         BRB     2$                                          ;
                                       62        04  AE  7D 0004F 1$:     MOVQ    NAME, (R2)                                  ; 4073
                                   10  BC            62  B0 00053 2$:     MOVW    (R2), @RET_LEN                              ; 4077
                                                         04 00057         RET                                                 ; 4080

; Routine Size:  88 bytes,     Routine Base:  CODE + 1954
```

```
1 %sbttl 'INSERT_FILENAME_BANNER - Get Name of the Current File'
1 !++
1 ! Functional Description:
1 !       This routine creates a banner phrase with the name of the current file.
1 !       Algorithm:
1 !               If the Filename, Type, and Version (FTV) fits on one line
1 !               If only one banner line exists....
1 !                                               insert FTV on only ONE line
1 !
1 !               If there are Three banner lines available...
1 !                                               insert Filename on one, Type on
1 !                                               another, and Version on third
1 !
1 !               Otherwise...                    insert Filename on one, Type and
1 !                                               Version on the second
1 !
1 ! Formal Parameters:
1 !       SCB             - Address of the SCB
1 !       STR_DESC        - Desc of String to Return
1 !       RET_LEN         - Return length of Desc.
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1 ROUTINE INSERT_FILENAME_BANNER (
1       SCB             : REF $BBLOCK,
1       STR_DESC        : REF VECTOR[2],
1       FRAME_PTR       : REF PAGE_ARRAY,
1       FRAME_WIDTH     ,               ! Number of Columns
1       FRAME_LENGTH                    ! Number of Rows
1       )  =
2 BEGIN
2 LITERAL
2     BIG_BANNER = 14,
2     LITTLE_BANNER = 7,
2     SMALL = 2,
2     LARGE = 1;
2
2 LOCAL
2     RET_LEN       : VECTOR[1],
2     PAGE_PTR      : REF PAGE_ARRAY,
2     SPACING       ,
2     CURRENT_PTR   ,
2     FTV_LEN       : VECTOR[1],
2     BANNER_TYPE   ,
2     BANNER_SIZE   ,
2     MAX_BAN_CHARS,
2     MAX_ROWS      ,
; 3218          4138 2       MAX_COLS          ,
; 3219          4139 2       NAME          : VECTOR[2],
; 3220          4140 2       TYPE          : VECTOR[2],
; 3221          4141 2       VERS          : VECTOR[2];
; 3222          4142 2
; 3223          4143 2   ! dont even try if there is no frame left
; 3224          4144 2   !
; 3225          4145 3   IF (.FRAME_LENGTH LSS 7)
; 3226          4146 2   THEN
; 3227          4147 2       RETURN 0;
; 3228          4148 2
; 3229          4149 2   PARSE_FILE_NAME (SCB[PSM$Q_FILE_SPECIFICATION], FSCN$_NAME, NAME);
; 3230          4150 2   PARSE_FILE_NAME (SCB[PSM$Q_FILE_SPECIFICATION], FSCN$_TYPE, TYPE);
; 3231          4151 2   PARSE_FILE_NAME (SCB[PSM$Q_FILE_SPECIFICATION], FSCN$_VERSION, VERS);
; 3232          4152 2
; 3233          4153 2   FTV_LEN[0] = .NAME[SIZE] + .TYPE[SIZE] + .VERS[SIZE];
; 3234          4154 2
; 3235          4155 2   BANNER_TYPE = BIG_BANNER;
; 3236          4156 2   BANNER_SIZE = LARGE;
; 3237          4157 2   SPACING = SMALL;                                    ! two spaces between banner rows
; 3238          4158 2
; 3239          4159 2   MAX_BAN_CHARS = (.FRAME_LENGTH/16) * (.FRAME_WIDTH/12);
; 3240          4160 2   MAX_ROWS = .BANNER_SIZE * (.FRAME_LENGTH / 16);
; 3241          4161 2   MAX_COLS = .BANNER_SIZE * (.FRAME_WIDTH / 12);
; 3242          4162 2
; 3243          4163 2   IF .MAX_BAN_CHARS LSS .FTV_LEN[0]
; 3244          4164 2   THEN
; 3245          4165 3       BEGIN
; 3246          4166 3       BANNER_SIZE = SMALL;
; 3247          4167 3       BANNER_TYPE = LITTLE_BANNER;
; 3248          4168 3       SPACING = LARGE;                                ! single space banner rows
; 3249          4169 3       MAX_ROWS = .BANNER_SIZE * (.FRAME_LENGTH / 18);
; 3250          4170 3       MAX_COLS = .BANNER_SIZE * (.FRAME_WIDTH / 14);
; 3251          4171 2       END;
; 3252          4172 2
; 3253          4173 2
; 3254          4174 2   ! Do somemore calculations to ensure consistent letter sizing
; 3255          4175 2   !         !X! Just to gte this out the door...  needs to be optimized later.rb
; 3256          4176 2   !
; 3257          4177 4   IF (     (.NAME[SIZE] GTR .MAX_COLS)
; 3258          4178 3                        OR
; 3259          4179 4            (.TYPE[SIZE] GTR .MAX_COLS)
; 3260          4180 3                        OR
; 3261          4181 4            (.VERS[SIZE] GTR .MAX_COLS)
; 3262          4182 3                        OR
; 3263          4183 4            (        (.MAX_ROWS LSS 3) AND
; 3264          4184 5            ((.TYPE[SIZE] + .VERS[SIZE]) GTR .MAX_COLS)
; 3265          4185 4                        OR
; 3266          4186 4            (.FTV_LEN[0] GTR (.MAX_COLS * .MAX_ROWS))  )
; 3267          4187 3            )
; 3268          4188 2   THEN
; 3269          4189 3       BEGIN
; 3270          4190 3       BANNER_SIZE = SMALL;
; 3271          4191 3       BANNER_TYPE = LITTLE_BANNER;
; 3272          4192 3       SPACING = LARGE;                                ! single space banner rows
; 3273          4193 3       MAX_ROWS = .BANNER_SIZE * (.FRAME_LENGTH / 18);
; 3274          4194 3       MAX_COLS = .BANNER_SIZE * (.FRAME_WIDTH / 14);
; 3275          4195  2     END;
; 3276          4196  2
; 3277          4197  2  ! Attempt to fit the filename, type, and version on one line
; 3278          4198  2  !
; 3279          4199  3 IF (.FTV_LEN[0] LEQ .MAX_COLS)                  ! insert on one line
; 3280          4200  2         AND                                     ! only if little banner
; 3281          4201  3   (.BANNER_TYPE EQL LITTLE_BANNER)
; 3282          4202  2 THEN
; 3283          4203  3     BEGIN
; 3284          4204  3     CURRENT_PTR = .STR_DESC[ADDR];
; 3285          4205  3     CURRENT_PTR = CH$MOVE(.NAME[SIZE], .NAME[ADDR], .CURRENT_PTR);
; 3286          4206  3     CURRENT_PTR = CH$MOVE(.TYPE[SIZE], .TYPE[ADDR], .CURRENT_PTR);
; 3287          4207  3     CURRENT_PTR = CH$MOVE(.VERS[SIZE], .VERS[ADDR], .CURRENT_PTR);
; 3288          4208  3     STR_DESC[SIZE] = .FTV_LEN[0];
; 3289          4209  3
; 3290          4210  3     INSERT_NAME_BANNER (.SCB,
; 3291          4211  3                         STR_DESC[SIZE],         ! job name desc
; 3292          4212  3                         FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
; 3293          4213  3                                                 ! ref to frame
; 3294          4214  3                         .FRAME_WIDTH,           ! max width Bann
; 3295          4215  3                         .BANNER_TYPE,           ! frame length
; 3296          4216  3                         .BANNER_TYPE);          ! max hght Bann str
; 3297          4217  3
; 3298          4218  3     RETURN  .BANNER_TYPE;                       ! return how much space used
; 3299          4219  3     END
; 3300          4220  2 ELSE
; 3301          4221  3     BEGIN
; 3302          4222  3     !  Move filename with truncated banners when not enough space
; 3303          4223  3     !
; 3304          4224  4     IF (        (.MAX_ROWS LEQ 1) )
; 3305          4225  4 !
; 3306          4226  4 ! !X! Comment this out .... causes too many filenames to be printed on
; 3307          4227  4 ! !X! a single line when two lines would be more appropriate.
; 3308          4228  4 !
; 3309          4229  4 !                       OR
; 3310          4230  4 !       (.NAME[SIZE] GTR .MAX_COLS)
; 3311          4231  4 !                       OR
; 3312          4232  4 !       (.TYPE[SIZE] GTR .MAX_COLS)
; 3313          4233  4 !                       OR
; 3314          4234  4 !       (.VERS[SIZE] GTR .MAX_COLS)
; 3315          4235  4 !                       OR
; 3316          4236  4 !       (       (.MAX_ROWS LSS 3) AND
; 3317          4237  4 !       ((.TYPE[SIZE] + .VERS[SIZE]) GTR .MAX_COLS)
; 3318          4238  4 !                       OR
; 3319          4239  4 !       (.FTV_LEN[0] GTR (.MAX_COLS * .MAX_ROWS))  )
; 3320          4240  4 !       )
; 3321          4241  3     THEN
; 3322          4242  4         BEGIN
; 3323          4243  4         CURRENT_PTR = .STR_DESC[ADDR];
; 3324          4244  4         CURRENT_PTR = CH$MOVE(.NAME[SIZE], .NAME[ADDR], .CURRENT_PTR);
; 3325          4245  4         CURRENT_PTR = CH$MOVE(.TYPE[SIZE], .TYPE[ADDR], .CURRENT_PTR);
; 3326          4246  4         CURRENT_PTR = CH$MOVE(.VERS[SIZE], .VERS[ADDR], .CURRENT_PTR);
; 3327          4247  4
; 3328          4248  4  !X! This is a cludge to get this out the door.  I will declare a valid
; 3329          4249  4 !   descriptor in the future and use LIB$TRIM_FILESPEC.
; 3330          4250  4 !
; 3331          4251  4         STR_DESC[SIZE] = .FTV_LEN[0];
```

```
4
4       IF .NAME[0] GEQ .MAX_COLS
4       THEN
5           (STR_DESC[SIZE] = .MAX_COLS)
4       ELSE
5           (IF .NAME[0]+.TYPE[0] GTR .MAX_COLS
5           THEN
5               STR_DESC[SIZE] = .NAME[0]
5               ELSE
6                   IF ((.NAME[0]+TYPE[0]+VERS[0]) GTR .MAX_COLS)
5                   THEN
4                       STR_DESC[SIZE] = .NAME[0]+.TYPE[0]);
4
4 !     IF .FTV_LEN[0] GTR (.MAX_ROWS * .MAX_COLS)
4 !     THEN
4 !         !  Trim the file spec to fit.
4 !         !
4 !         LIB$TRIM_FILESPEC ( STR_DESC, STR_DESC[0],
4 !                         .MAX_COLS,
4 !                         STR_DESC[SIZE]);
4
4
4           RET_LEN[0] = INSERT_NAME_BANNER (
4                               .SCB,
4                               STR_DESC[SIZE], ! job name desc
4                               FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
4                                         ! ref to frame
4                               .FRAME_WIDTH,    ! max width Bann
4                               .BANNER_TYPE,    ! frame length
4                               .BANNER_TYPE);   ! max hght Bann str
4
4       RETURN  .RET_LEN[0];     ! return how much space used
4       END
3   ELSE        ! Should be able to insert it... Make it pretty
4       BEGIN
4       IF .MAX_ROWS GEQ 3
4       THEN
5           BEGIN
5
5           PAGE_PTR = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]];
5           INSERT_NAME_BANNER (
5                       .SCB,
5                       NAME[SIZE],                         ! file name
5                       PAGE_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
5                                                           ! ref to frame
5                       .FRAME_WIDTH,                       ! max width Bann
5                       .BANNER_TYPE,                       ! frame length
5                       .BANNER_TYPE);                      ! max hght Bann str
5
5           PAGE_PTR =  PAGE_PTR[0,(.BANNER_TYPE+.SPACING),
5                                .SCB[PSM$L_PAGE_WIDTH]];
5           INSERT_NAME_BANNER (
5                       .SCB,
5                       TYPE[SIZE],                         ! file type
5                       PAGE_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
5                                                           ! ref to frame
5                       .FRAME_WIDTH,                       ! max width Bann
5                               .BANNER_TYPE,                           ! frame length
5                               .BANNER_TYPE);                          ! max hght Bann str
5
5                   PAGE_PTR =  PAGE_PTR[0,(.BANNER_TYPE+.SPACING),
5                                           .SCB[PSM$L_PAGE_WIDTH]];
5                   INSERT_NAME_BANNER (
5                               .SCB,
5                               VERS[SIZE],                             ! file version number
5                               PAGE_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
5                                                                       ! ref to frame
5                               .FRAME_WIDTH,                           ! max width Bann
5                               .BANNER_TYPE,                           ! frame length
5                               .BANNER_TYPE);                          ! max hght Bann str
5
5                   RETURN  (3 * (.BANNER_TYPE + .SPACING));            ! return count of used
5                                                                       !  space
5                   END
4               ELSE
5                   BEGIN
5                   CURRENT_PTR = .STR_DESC[ADDR];
5                   CURRENT_PTR = CH$MOVE(.TYPE[SIZE], .TYPE[ADDR], .CURRENT_PTR);
5                   CURRENT_PTR = CH$MOVE(.VERS[SIZE], .VERS[ADDR], .CURRENT_PTR);
5                   STR_DESC[SIZE] = .TYPE[SIZE] + .VERS[SIZE];
5
5                   PAGE_PTR = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]];
5                   INSERT_NAME_BANNER (
5                               .SCB,
5                               NAME[SIZE],                             ! file name
5                               PAGE_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
5                                                                       ! ref to frame
5                               .FRAME_WIDTH,                           ! max width Bann
5                               .BANNER_TYPE,                           ! frame length
5                               .BANNER_TYPE);                          ! max hght Bann str
5
5                   PAGE_PTR =  PAGE_PTR[0,(.BANNER_TYPE+.SPACING),
5                                           .SCB[PSM$L_PAGE_WIDTH]];
5                   INSERT_NAME_BANNER (
5                               .SCB,
5                               STR_DESC[SIZE],                         ! file and version type
5                               PAGE_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
5                                                                       ! ref to frame
5                               .FRAME_WIDTH,                           ! max width Bann
5                               .BANNER_TYPE,                           ! frame length
5                               .BANNER_TYPE);                          ! max hght Bann str
5
5                   RETURN (2 * (.BANNER_TYPE + .SPACING));             ! return count of used
5                                                                       !  space
4                   END;
3               END;
2           END;
1   END;



                                        0FFC 00000 INSERT_FILENAME_BANNER:
```

```
                                                         .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11                 ; 4114
                5E                 20  C2 00002          SUBL2   #32, SP
                07          14     AC  D1 00005          CMPL    FRAME_LENGTH, #7                                     ; 4145
                                   03  18 00009          BGEQ    1$
                                 0211  31 0000B          BRW     12$
                            18     AE  9F 0000E 1$:      PUSHAB  NAME                                                 ; 4149
                                   06  DD 00011          PUSHL   #6
     50         04     AC 00000098 8F  C1 00013          ADDL3   #152, SCB, R0
                       52          60  9E 0001C          MOVAB   (R0), R2
                                   52  DD 0001F          PUSHL   R2
             EC5F      CF          03  FB 00021          CALLS   #3, PARSE_FILE_NAME
                            10     AE  9F 00026          PUSHAB  TYPE                                                 ; 4150
                                   07  DD 00029          PUSHL   #7
                                   52  DD 0002B          PUSHL   R2
             EC53      CF          03  FB 0002D          CALLS   #3, PARSE_FILE_NAME
                            08     AE  9F 00032          PUSHAB  VERS                                                 ; 4151
                                   08  DD 00035          PUSHL   #8
                                   52  DD 00037          PUSHL   R2
             EC47      CF          03  FB 00039          CALLS   #3, PARSE_FILE_NAME
                       5B   18     AE  D0 0003E          MOVL    NAME, R11                                            ; 4153
                04     AE   10 BE4B    9E 00042          MOVAB   @TYPE[R11], 4(SP)
     6E         04     AE   08     AE  C1 00048          ADDL3   VERS, 4(SP), FTV_LEN
                       57          0E  D0 0004E          MOVL    #14, BANNER_TYPE                                     ; 4155
                       50          01  D0 00051          MOVL    #1, BANNER_SIZE                                      ; 4156
                       54          02  D0 00054          MOVL    #2, SPACING                                          ; 4157
     52         14     AC          10  C7 00057          DIVL3   #16, FRAME_LENGTH, R2                                ; 4159
                       5A   10     AC  D0 0005C          MOVL    FRAME_WIDTH, R10
     51                5A          0C  C7 00060          DIVL3   #12, R10, R1
     53                52          51  C5 00064          MULL3   R1, R2, MAX_BAN_CHARS
     59                50          52  C5 00068          MULL3   R2, BANNER_SIZE, MAX_ROWS                            ; 4160
     56                50          51  C5 0006C          MULL3   R1, BANNER_SIZE, MAX_COLS                            ; 4161
                       6E          53  D1 00070          CMPL    MAX_BAN_CHARS, FTV_LEN                               ; 4163
                                   1A  18 00073          BGEQ    2$
                       50          02  D0 00075          MOVL    #2, BANNER_SIZE                                      ; 4166
                       57          07  D0 00078          MOVL    #7, BANNER_TYPE                                      ; 4167
                       54          01  D0 0007B          MOVL    #1, SPACING                                          ; 4168
     51         14     AC          12  C7 0007E          DIVL3   #18, FRAME_LENGTH, R1                                ; 4169
     59                51          50  C5 00083          MULL3   BANNER_SIZE, R1, MAX_ROWS
     51                5A          0E  C7 00087          DIVL3   #14, R10, R1                                         ; 4170
     56                51          50  C5 0008B          MULL3   BANNER_SIZE, R1, MAX_COLS
                       56          5B  D1 0008F 2$:      CMPL    R11, MAX_COLS                                        ; 4177
                                   25  14 00092          BGTR    4$
                       56   10     AE  D1 00094          CMPL    TYPE, MAX_COLS                                       ; 4179
                                   1F  14 00098          BGTR    4$
                       56   08     AE  D1 0009A          CMPL    VERS, MAX_COLS                                       ; 4181
                                   19  14 0009E          BGTR    4$
                       03          59  D1 000A0          CMPL    MAX_ROWS, #3                                         ; 4183
                                   0B  18 000A3          BGEQ    3$
     51         10     AE   08     AE  C1 000A5          ADDL3   VERS, TYPE, R1                                       ; 4184
                       56          51  D1 000AB          CMPL    R1, MAX_COLS
                                   09  14 000AE          BGTR    4$
     51                56          59  C5 000B0 3$:      MULL3   MAX_ROWS, MAX_COLS, R1                               ; 4186
                       51          6E  D1 000B4          CMPL    FTV_LEN, R1
                                   1A  15 000B7          BLEQ    5$
                       50          02  D0 000B9 4$:      MOVL    #2, BANNER_SIZE                                      ; 4190
                       57          07  D0 000BC          MOVL    #7, BANNER_TYPE                                      ; 4191
                       54          01  D0 000BF          MOVL    #1, SPACING                                          ; 4192
```

```
51       14  AC            12 C7 000C2       DIVL3   #18, FRAME_LENGTH, R1               ; 4193
59           51            50 C5 000C7       MULL3   BANNER_SIZE, R1, MAX_ROWS           ;
51           5A            0E C7 000CB       DIVL3   #14, R10, R1                        ; 4194
56           51            50 C5 000CF       MULL3   BANNER_SIZE, R1, MAX_COLS           ;
             56            6E D1 000D3 5$:   CMPL    FTV_LEN, MAX_COLS                   ; 4199
                           38 14 000D6       BGTR    6$                                  ;
             07            57 D1 000D8       CMPL    BANNER_TYPE, #7                     ; 4201
                           33 12 000DB       BNEQ    6$                                  ;
             58       08   AC D0 000DD       MOVL    STR_DESC, R8                        ; 4204
             53       04   A8 D0 000E1       MOVL    4(R8), CURRENT_PTR                  ;
63       1C  BE            5B 28 000E5       MOVC3   R11, @NAME+4, (CURRENT_PTR)         ; 4205
63       14  BE       10   AE 28 000EA       MOVC3   TYPE, @TYPE+4, (CURRENT_PTR)        ; 4206
63       0C  BE       08   AE 28 000F0       MOVC3   VERS, @VERS+4, (CURRENT_PTR)        ; 4207
             68            6E D0 000F6       MOVL    FTV_LEN, (R8)                       ; 4208
                           57 DD 000F9       PUSHL   BANNER_TYPE                         ; 4216
                           57 DD 000FB       PUSHL   BANNER_TYPE                         ; 4215
                           5A DD 000FD       PUSHL   R10                                 ; 4214
                      0C   AC DD 000FF       PUSHL   FRAME_PTR                           ; 4212
                           58 DD 00102       PUSHL   R8                                  ;
                      04   AC DD 00104       PUSHL   SCB                                 ;
      0000V  CF            06 FB 00107       CALLS   #6, INSERT_NAME_BANNER              ;
             50            57 D0 0010C       MOVL    BANNER_TYPE, R0                     ; 4221
                              04 0010F       RET                                         ;
             01            59 D1 00110 6$:   CMPL    MAX_ROWS, #1                        ; 4224
                           5A 14 00113       BGTR    10$                                 ;
             58       08   AC D0 00115       MOVL    STR_DESC, R8                        ; 4243
             53       04   A8 D0 00119       MOVL    4(R8), CURRENT_PTR                  ;
63       1C  BE            5B 28 0011D       MOVC3   R11, @NAME+4, (CURRENT_PTR)         ; 4244
63       14  BE       10   AE 28 00122       MOVC3   TYPE, @TYPE+4, (CURRENT_PTR)        ; 4245
63       0C  BE       08   AE 28 00128       MOVC3   VERS, @VERS+4, (CURRENT_PTR)        ; 4246
             68            6E D0 0012E       MOVL    FTV_LEN, (R8)                       ; 4251
             56            5B D1 00131       CMPL    R11, MAX_COLS                       ; 4253
                           05 19 00134       BLSS    7$                                  ;
             68            56 D0 00136       MOVL    MAX_COLS, (R8)                      ; 4255
                           20 11 00139       BRB     9$                                  ;
             56       04   AE D1 0013B 7$:   CMPL    4(SP), MAX_COLS                     ; 4257
                           05 15 0013F       BLEQ    8$                                  ;
             68            5B D0 00141       MOVL    R11, (R8)                           ; 4259
                           15 11 00144       BRB     9$                                  ;
             50       10   AE 9E 00146 8$:   MOVAB   TYPE, R0                            ; 4261
             50       08 AE40 9E 0014A       MOVAB   VERS[R0], R0                        ;
             50            5B C0 0014F       ADDL2   R11, R0                             ;
             56            50 D1 00152       CMPL    R0, MAX_COLS                        ;
                           04 15 00155       BLEQ    9$                                  ;
             68       04   AE D0 00157       MOVL    4(SP), (R8)                         ; 4263
                           57 DD 0015B 9$:   PUSHL   BANNER_TYPE                         ; 4281
                           57 DD 0015D       PUSHL   BANNER_TYPE                         ; 4280
                           5A DD 0015F       PUSHL   R10                                 ; 4279
                      0C   AC DD 00161       PUSHL   FRAME_PTR                           ; 4277
                           58 DD 00164       PUSHL   R8                                  ;
                      04   AC DD 00166       PUSHL   SCB                                 ;
      0000V  CF            06 FB 00169       CALLS   #6, INSERT_NAME_BANNER              ;
                              04 0016E       RET                                         ; 4286
             56       0C   AC D0 0016F 10$:  MOVL    FRAME_PTR, PAGE_PTR                 ; 4291
58           57            54 C1 00173       ADDL3   SPACING, BANNER_TYPE, R8            ; 4301
50       04  AC 00000200   8F C1 00177       ADDL3   #512, SCB, R0                       ; 4302
             5B            60 9E 00180       MOVAB   (R0), R11                           ;
```

```
                   03          59  D1 00183         CMPL    MAX_ROWS, #3                                       ; 4287
                               4C  19 00186         BLSS    11$                                                 :
                               57  DD 00188         PUSHL   BANNER_TYPE                                        ; 4299
                               57  DD 0018A         PUSHL   BANNER_TYPE                                        ; 4298
                        0440   8F  BB 0018C         PUSHR   #^M<R6,R10>                                        ; 4295
                          28   AE  9F 00190         PUSHAB  NAME                                               ; 4294
                          04   AC  DD 00193         PUSHL   SCB                                                ; 4295
            0000V  CF          06  FB 00196         CALLS   #6, INSERT_NAME_BANNER                              :
      50           58          6B  C5 0019B         MULL3   (R11), R8, R0                                      ; 4302
                   56          50  C0 0019F         ADDL2   R0, PAGE_PTR                                        :
                               57  DD 001A2         PUSHL   BANNER_TYPE                                        ; 4310
                               57  DD 001A4         PUSHL   BANNER_TYPE                                        ; 4309
                        0440   8F  BB 001A6         PUSHR   #^M<R6,R10>                                        ; 4306
                          20   AE  9F 001AA         PUSHAB  TYPE                                               ; 4305
                          04   AC  DD 001AD         PUSHL   SCB                                                ; 4306
            0000V  CF          06  FB 001B0         CALLS   #6, INSERT_NAME_BANNER                              :
      50           58          6B  C5 001B5         MULL3   (R11), R8, R0                                      ; 4313
                   56          50  C0 001B9         ADDL2   R0, PAGE_PTR                                        :
                               57  DD 001BC         PUSHL   BANNER_TYPE                                        ; 4321
                               57  DD 001BE         PUSHL   BANNER_TYPE                                        ; 4320
                        0440   8F  BB 001C0         PUSHR   #^M<R6,R10>                                        ; 4317
                          18   AE  9F 001C4         PUSHAB  VERS                                               ; 4316
                          04   AC  DD 001C7         PUSHL   SCB                                                ; 4317
            0000V  CF          06  FB 001CA         CALLS   #6, INSERT_NAME_BANNER                              :
      50           58          03  C5 001CF         MULL3   #3, R8, R0                                         ; 4323
                                   04 001D3         RET                                                        ; 4327
                   59     08   AC  D0 001D4 11$:    MOVL    STR_DESC, R9                                       ; 4328
                   53     04   A9  D0 001D8         MOVL    4(R9), CURRENT_PTR                                  :
      63     14    BE     10   AE  28 001DC         MOVC3   TYPE, @TYPE+4, (CURRENT_PTR)                       ; 4329
      63     0C    BE     08   AE  28 001E2         MOVC3   VERS, @VERS+4, (CURRENT_PTR)                       ; 4330
      69     10    AE     08   AE  C1 001E8         ADDL3   VERS, TYPE, (R9)                                   ; 4331
                               57  DD 001EE         PUSHL   BANNER_TYPE                                        ; 4341
                               57  DD 001F0         PUSHL   BANNER_TYPE                                        ; 4340
                        0440   8F  BB 001F2         PUSHR   #^M<R6,R10>                                        ; 4337
                          28   AE  9F 001F6         PUSHAB  NAME                                               ; 4336
                          04   AC  DD 001F9         PUSHL   SCB                                                ; 4337
            0000V  CF          06  FB 001FC         CALLS   #6, INSERT_NAME_BANNER                              :
      50           58          6B  C5 00201         MULL3   (R11), R8, R0                                      ; 4344
                   56          50  C0 00205         ADDL2   R0, PAGE_PTR                                        :
                               57  DD 00208         PUSHL   BANNER_TYPE                                        ; 4352
                               57  DD 0020A         PUSHL   BANNER_TYPE                                        ; 4351
                        0440   8F  BB 0020C         PUSHR   #^M<R6,R10>                                        ; 4348
                               59  DD 00210         PUSHL   R9                                                  :
                          04   AC  DD 00212         PUSHL   SCB                                                 :
            0000V  CF          06  FB 00215         CALLS   #6, INSERT_NAME_BANNER                              :
      50           58          01  78 0021A         ASHL    #1, R8, R0                                         ; 4354
                                   04 0021E         RET                                                        ; 4221
                               50  D4 0021F 12$:    CLRL    R0                                                 ; 4359
                                   04 00221         RET                                                         :

; Routine Size:  546 bytes,    Routine Base:  CODE + 19AC
```

```
  1 %sbttl 'INSERT_JOBNUMBER_BANNER - Get Job Number of the current Job'
  1 !++
  1 ! Functional Description:
  1 !         This routine creates a banner phrase with the Job Number
  1 !
  1 ! Formal Parameters:
  1 !         SCB             - Address of the SCB
  1 !         STR_DESC        - Desc of String to Return
  1 !         RET_LEN         - Return length of Desc.
  1 !
  1 ! Implicit Inputs:
  1 !                 none
  1 !
  1 ! Implicit Outputs:
  1 !                 none
  1 !
  1 ! Returned Value:
  1 !                 none
  1 !
  1 ! Side Effects:
  1 !                 none
  1 !--
  1 ROUTINE INSERT_JOBNUMBER_BANNER (
  1         SCB             : REF $BBLOCK,
  1         STR_DESC        : REF VECTOR[2],
  1         FRAME_PTR       : REF PAGE_ARRAY,
  1         FRAME_WIDTH     ,                       ! Number of Columns
  1         FRAME_LENGTH                            ! Number of Rows
  1         )  =
  2 BEGIN
  2 BIND
P 2     SENT_FORMAT = $DESCRIPTOR (
P 2         'JOB ',
  2         '!UL'),
  2
P 2     NUM_FORMAT = $DESCRIPTOR (
  2         '!UL');
  2
  2 LITERAL
  2     LITTLE_BANNER = 7,
  2     SMALL = 2,
  2     LARGE = 1;
  2
  2 LOCAL
  2     RET_LEN         : VECTOR[1],
  2     PAGE_PTR        : REF PAGE_ARRAY,
  2     CURRENT_PTR     ,
  2     JOB_LEN         : INITIAL (0),
  2     BANNER_TYPE     ,
  2     SPACING         ,
  2     BANNER_SIZE     ,
  2     MAX_BAN_CHARS,
  2     BUFFER          : VECTOR[10,byte],
  2     MAX_COLS        ,
  2     NUMBER          : VECTOR[2];
  2
  2 NUMBER[SIZE] = %ALLOCATION(BUFFER);
  2 NUMBER[ADDR] = BUFFER;
  2
  2 ! dont even try if there is no frame left
  2 !
  3 IF (.FRAME_LENGTH LSS 7)
  2 THEN
  2     RETURN 0;
  2
  2 MAX_BAN_CHARS = (.FRAME_LENGTH/16) * (.FRAME_WIDTH/12);
  2
  2 BANNER_SIZE = SMALL;
  2 BANNER_TYPE = LITTLE_BANNER;
  2 SPACING = LARGE;                                ! single space banner rows
  2
  2 MAX_COLS = .BANNER_SIZE * (.FRAME_WIDTH / 12);
  2
  2
P 2 $FAO (  NUM_FORMAT,
P 2         JOB_LEN,                                ! str[size] > fetched namelen
P 2         NUMBER[0],
P 2         .SCB[PSM$L_ENTRY_NUMBER]                ! job number
  2      );
  2
  2 IF (.JOB_LEN+4) LEQ .MAX_COLS        ! insert job&num on one line
  2 THEN
  3     BEGIN
P 3     $FAO (  SENT_FORMAT,
P 3             JOB_LEN,
P 3             STR_DESC[0],
P 3             .SCB[PSM$L_ENTRY_NUMBER]            ! job number
  3          );
  3     STR_DESC[SIZE] = .JOB_LEN;                  ! update the size
  3     END
  2 ELSE                                            ! copy the string.. num only
  3     BEGIN
  3     IF .JOB_LEN LEQ .MAX_COLS
  3     THEN
  4         BEGIN
  4         CURRENT_PTR = .STR_DESC[ADDR];
  4         CURRENT_PTR = CH$MOVE(.JOB_LEN, .NUMBER[ADDR], .CURRENT_PTR);
  4         STR_DESC[SIZE] = .JOB_LEN;
  4         END
  3     ELSE
  3         RETURN 0;                               ! exit ... No Room
  2     END;
  2
  2 INSERT_NAME_BANNER (
  2                         .SCB,
  2                         STR_DESC[SIZE],         ! job name desc
  2                         FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
  2                                                 ! ref to frame
  2                         .FRAME_WIDTH,           ! max width Bann
  2                         .BANNER_TYPE,           ! frame length
  2                         .BANNER_TYPE);          ! max hght Bann str
  2
  2 RETURN  .BANNER_TYPE;                           ! return how much space in
  2                                                 !  length used
```

```
; 3555        4474  1 END;



                20  42  4F  4A  01BCE P.ACK:   .ASCII   \JOB \
                    4C  55  21  01BD2          .ASCII   \!UL\                                       ;
                                01BD5          .BLKB    3
                      00000007  01BD8 P.ACJ:   .LONG    7
                      00000000' 01BDC          .ADDRESS P.ACK                                       ;
                    4C  55  21  01BE0 P.ACM:   .ASCII   \!UL\                                       ;
                                01BE3          .BLKB    1
                      00000003  01BE4 P.ACL:   .LONG    3                                           ;
                      00000000' 01BE8          .ADDRESS P.ACM                                       ;

                                      SENT_FORMAT=      P.ACJ
                                      NUM_FORMAT=       P.ACL


                          03FC  00000 INSERT_JOBNUMBER_BANNER:
                                               .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9                ; 4382
          59 00000000G  00  9E  00002          MOVAB    SYS$FAO, R9                                 ;
        5E              14  C2  00009          SUBL2    #20, SP                                     ;
                        7E  D4  0000C          CLRL     JOB_LEN                                     ; 4389
        04  AE          0A  D0  0000E          MOVL     #10, NUMBER                                 ; 4416
        08  AE      0C  AE  9E  00012          MOVAB    BUFFER, NUMBER+4                            ; 4417
        07          14  AC  D1  00017          CMPL     FRAME_LENGTH, #7                            ; 4421
                        76  19  0001B          BLSS     3$                                          ;
51      14  AC          10  C7  0001D          DIVL3    #16, FRAME_LENGTH, R1                       ; 4425
50      10  AC          0C  C7  00022          DIVL3    #12, FRAME_WIDTH, R0                        ;
            51          50  C4  00027          MULL2    R0, MAX_BAN_CHARS                           ;
            51          02  D0  0002A          MOVL     #2, BANNER_SIZE                             ; 4427
            58          07  D0  0002D          MOVL     #7, BANNER_TYPE                             ; 4428
            52          01  D0  00030          MOVL     #1, SPACING                                 ; 4429
52          51          50  C5  00033          MULL3    R0, BANNER_SIZE, MAX_COLS                   ; 4431
            57      04  AC  D0  00037          MOVL     SCB, R7                                     ; 4438
                    58  A7  DD  0003B          PUSHL    88(R7)                                      ;
                    08  AE  9F  0003E          PUSHAB   NUMBER                                      ;
                    08  AE  9F  00041          PUSHAB   JOB_LEN                                     ;
                    B1  AF  9F  00044          PUSHAB   NUM_FORMAT                                  ;
            69          04  FB  00047          CALLS    #4, SYS$FAO                                 ;
50          6E          04  C1  0004A          ADDL3    #4, JOB_LEN, R0                             ; 4440
            52          50  D1  0004E          CMPL     R0, MAX_COLS                                ;
                        15  14  00051          BGTR     1$                                          ;
                    58  A7  DD  00053          PUSHL    88(R7)                                      ; 4447
                    08  AC  DD  00056          PUSHL    STR_DESC                                    ;
                    08  AE  9F  00059          PUSHAB   JOB_LEN                                     ;
                    8D  AF  9F  0005C          PUSHAB   SENT_FORMAT                                 ;
            69          04  FB  0005F          CALLS    #4, SYS$FAO                                 ;
        08  BC          6E  D0  00062          MOVL     JOB_LEN, @STR_DESC                          ; 4448
                        15  11  00066          BRB      2$                                          ; 4440
            52          6E  D1  00068 1$:      CMPL     JOB_LEN, MAX_COLS                           ; 4452
                        26  14  0006B          BGTR     3$                                          ;
            56      08  AC  D0  0006D          MOVL     STR_DESC, R6                                ; 4455
            53      04  A6  D0  00071          MOVL     4(R6), CURRENT_PTR                          ;
63      08  BE          6E  28  00075          MOVC3    JOB_LEN, @NUMBER+4, (CURRENT_PTR)           ; 4456
            66          6E  D0  0007A          MOVL     JOB_LEN, (R6)                               ; 4457
                        58  DD  0007D 2$:      PUSHL    BANNER_TYPE                                 ; 4470
```

```
                                          58  DD 0007F            PUSHL   BANNER_TYPE                          ; 4469
                      7E          0C      AC  7D 00081            MOVQ    FRAME_PTR, -(SP)                     ; 4466
                                  08      AC  DD 00085            PUSHL   STR_DESC                             ;
                                          57  DD 00088            PUSHL   R7                                   ;
                0000V CF                  06  FB 0008A            CALLS   #6, INSERT_NAME_BANNER               ;
                      50                  58  D0 0008F            MOVL    BANNER_TYPE, R0                      ; 4472
                                              04 00092            RET                                          ;
                                          50  D4 00093 3$:        CLRL    R0                                   ; 4474
                                              04 00095            RET                                          ;

; Routine Size:  150 bytes,    Routine Base:  CODE + 1BEC
; 3557          4475  1 %sbttl 'GET_JOB_NAME - Get Name of the Current Job'
; 3558          4476  1 !++
; 3559          4477  1 ! Functional Description:
; 3560          4478  1 !       This routine creates a phrase with the name of the current job.
; 3561          4479  1 !
; 3562          4480  1 ! Formal Parameters:
; 3563          4481  1 !       SCB             - Address of the SCB
; 3564          4482  1 !       STR_DESC        - Desc of String to Return
; 3565          4483  1 !       RET_LEN         - Return length of Desc.
; 3566          4484  1 !
; 3567          4485  1 ! Implicit Inputs:
; 3568          4486  1 !               none
; 3569          4487  1 !
; 3570          4488  1 ! Implicit Outputs:
; 3571          4489  1 !               none
; 3572          4490  1 !
; 3573          4491  1 ! Returned Value:
; 3574          4492  1 !               none
; 3575          4493  1 !
; 3576          4494  1 ! Side Effects:
; 3577          4495  1 !               none
; 3578          4496  1 !--
; 3579          4497  1 ROUTINE GET_JOB_NAME (
; 3580          4498  1               SCB             : REF $BBLOCK,          ! SCB
; 3581          4499  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 3582          4500  1               RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 3583          4501  1                       )       : NOVALUE =
; 3584          4502  2 BEGIN
; 3585          4503  2 BIND
; 3586        P 4504  2     SENT80_FORMAT = $DESCRIPTOR (
; 3587          4505  2             '!AS');
; 3588          4506  2
; 3589        P 4507  2     $FAO (  SENT80_FORMAT,
; 3590        P 4508  2             RET_LEN[0],                             ! str[size] > fetched namelen
; 3591        P 4509  2             STR_DESC[0],
; 3592        P 4510  2             SCB[PSM$Q_JOB_NAME],                    ! job name
; 3593          4511  2         );
; 3594          4512  2
; 3595          4513  2 RETURN SS$_NORMAL;
; 3596          4514  1 END;


                                53  41  21   01C82 P.ACO:   .ASCII  \!AS\                                   ;
                                             01C85          .BLKB   3
                                  00000003   01C88 P.ACN:   .LONG   3                                       ;
                                  00000000'  01C8C          .ADDRESS P.ACO                                  ;

                                                     SENT80_FORMAT=         P.ACN


                                        0000 00000 GET_JOB_NAME:
                                                            .WORD   Save nothing                            ; 4497
                7E        04   AC 000000A8    8F   C1 00002         ADDL3   #168, SCB, -(SP)                        ; 4511
                                              08   AC   DD 0000B    PUSHL   STR_DESC                                ;
                                              0C   AC   DD 0000E    PUSHL   RET_LEN                                 ;
                                              E4   AF   9F 00011    PUSHAB  SENT80_FORMAT                           ;
```

```
                        00000000G  00          04  FB 00014          CALLS    #4, SYS$FAO                                  ;
                                                   04 0001B          RET                                                   ; 4514

; Routine Size:  28 bytes,     Routine Base:  CODE + 1C90
; 3598          4515  1 %sbttl 'GET_EOJ - Get the Phrase End of Job'
; 3599          4516  1 !++
; 3600          4517  1 ! Functional Description:
; 3601          4518  1 !       This routine creates a phrase with "EOJ" or "END OF JOB".
; 3602          4519  1 !
; 3603          4520  1 ! Formal Parameters:
; 3604          4521  1 !       SCB             - Address of the SCB
; 3605          4522  1 !       STR_DESC        - Desc of String to Return
; 3606          4523  1 !       RET_LEN         - Return length of Desc.
; 3607          4524  1 !
; 3608          4525  1 ! Implicit Inputs:
; 3609          4526  1 !               none
; 3610          4527  1 !
; 3611          4528  1 ! Implicit Outputs:
; 3612          4529  1 !               none
; 3613          4530  1 !
; 3614          4531  1 ! Returned Value:
; 3615          4532  1 !               none
; 3616          4533  1 !
; 3617          4534  1 ! Side Effects:
; 3618          4535  1 !               none
; 3619          4536  1 !--
; 3620          4537  1 ROUTINE GET_EOJ (
; 3621          4538  1               SCB             : REF $BBLOCK,          ! SCB
; 3622          4539  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 3623          4540  1               RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 3624          4541  1                       )       : NOVALUE =
; 3625          4542  2 BEGIN
; 3626          4543  2 BIND
; 3627        P 4544  2     SENT132_FORMAT = $DESCRIPTOR (
; 3628          4545  2         'END OF JOB'),                        ! -
; 3629          4546  2
; 3630        P 4547  2     SENT80_FORMAT = $DESCRIPTOR (
; 3631          4548  2         'EOJ');
; 3632          4549  2
; 3633        P 4550  2 $FAO (  SENT132_FORMAT,
; 3634        P 4551  2         RET_LEN[0],                           ! return length
; 3635        P 4552  2         STR_DESC[0],                          ! address of string
; 3636          4553  2     );
; 3637          4554  2
; 3638          4555  2 ! Is it short enough to allow the words "End of Job" to be printed ?
; 3639          4556  2 !
; 3640          4557  3 IF ((12 * .RET_LEN[0]) GTR .SCB[PSM$L_PAGE_WIDTH])
; 3641          4558  2 THEN
; 3642        P 4559  2     $FAO (  SENT80_FORMAT,
; 3643        P 4560  2             RET_LEN[0],                       ! str[size] > fetched namelen
; 3644        P 4561  2             STR_DESC[0],
; 3645          4562  2         );
; 3646          4563  2
; 3647          4564  2 RETURN SS$_NORMAL;
; 3648          4565  1 END;


                42  4F  4A  20  46  4F  20  44  4E  45  01CAC P.ACQ:  .ASCII  \END OF JOB\                                    ;
                                                        01CB6          .BLKB   2
                                              0000000A  01CB8 P.ACP:  .LONG   10                                              ;
```

```
                              00000000' 01CBC          .ADDRESS P.ACQ                          ;
                           4A  4F  45 01CC0 P.ACS:     .ASCII  \EOJ\                           ;
                                      01CC3            .BLKB   1
                              00000003 01CC4 P.ACR:    .LONG   3                               ;
                              00000000' 01CC8          .ADDRESS P.ACS                          ;

                                             SENT132_FORMAT=       P.ACP
                                             SENT80_FORMAT=        P.ACR


                                 0004 00000 GET_EOJ:.WORD    Save R2                             ; 4537
                 52 00000000G  00   9E 00002         MOVAB   SYS$FAO, R2                         ;
                          08   AC   DD 00009         PUSHL   STR_DESC                            ; 4553
                          0C   AC   DD 0000C         PUSHL   RET_LEN                             ;
                          DA   AF   9F 0000F         PUSHAB  SENT132_FORMAT                      ;
                   62          03   FB 00012         CALLS   #3, SYS$FAO                         ;
                   51     0C   BC   3C 00015         MOVZWL  @RET_LEN, R1                        ; 4557
                   51          0C   C4 00019         MULL2   #12, R1                             ;
                   50     04   AC   D0 0001C         MOVL    SCB, R0                             ;
              0200 C0          51   D1 00020         CMPL    R1, 512(R0)                         ;
                               0C   15 00025         BLEQ    1$                                  ;
                          08   AC   DD 00027         PUSHL   STR_DESC                            ; 4562
                          0C   AC   DD 0002A         PUSHL   RET_LEN                             ;
                          C8   AF   9F 0002D         PUSHAB  SENT80_FORMAT                       ;
                   62          03   FB 00030         CALLS   #3, SYS$FAO                         ;
                               04 00033 1$:          RET                                         ; 4565

; Routine Size:  52 bytes,    Routine Base:  CODE + 1CCC
; 3650          4566  1 %sbttl 'GET_EOF - Get the Phrase End of File'
; 3651          4567  1 !++
; 3652          4568  1 ! Functional Description:
; 3653          4569  1 !       This routine creates a phrase with "EOF" or "END OF FILE".
; 3654          4570  1 !
; 3655          4571  1 ! Formal Parameters:
; 3656          4572  1 !       SCB             - Address of the SCB
; 3657          4573  1 !       STR_DESC        - Desc of String to Return
; 3658          4574  1 !       RET_LEN         - Return length of Desc.
; 3659          4575  1 !
; 3660          4576  1 ! Implicit Inputs:
; 3661          4577  1 !               none
; 3662          4578  1 !
; 3663          4579  1 ! Implicit Outputs:
; 3664          4580  1 !               none
; 3665          4581  1 !
; 3666          4582  1 ! Returned Value:
; 3667          4583  1 !               none
; 3668          4584  1 !
; 3669          4585  1 ! Side Effects:
; 3670          4586  1 !               none
; 3671          4587  1 !--
; 3672          4588  1 ROUTINE GET_EOF (
; 3673          4589  1               SCB             : REF $BBLOCK,          ! SCB
; 3674          4590  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 3675          4591  1               RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 3676          4592  1                       )       : NOVALUE =
; 3677          4593  2 BEGIN
; 3678          4594  2 BIND
; 3679        P 4595  2     SENT132_FORMAT = $DESCRIPTOR (
; 3680          4596  2         'END OF FILE'),                     ! -
; 3681          4597  2
; 3682        P 4598  2     SENT80_FORMAT = $DESCRIPTOR (
; 3683          4599  2         'EOF');
; 3684          4600  2
; 3685        P 4601  2 $FAO (  SENT132_FORMAT,
; 3686        P 4602  2         RET_LEN[0],                         ! return length
; 3687        P 4603  2         STR_DESC[0],                        ! address of string
; 3688          4604  2     );
; 3689          4605  2
; 3690          4606  2 ! Is it short enough to allow the words "End of Job" to be printed ?
; 3691          4607  2 !
; 3692          4608  3 IF ((12 * .RET_LEN[0]) GTR .SCB[PSM$L_PAGE_WIDTH])
; 3693          4609  2 THEN
; 3694        P 4610  2     $FAO (  SENT80_FORMAT,
; 3695        P 4611  2             RET_LEN[0],                             ! str[size] > fetched namelen
; 3696        P 4612  2             STR_DESC[0],
; 3697          4613  2         );
; 3698          4614  2
; 3699          4615  2 RETURN SS$_NORMAL;
; 3700          4616  1 END;


        45  4C  49  46  20  46  4F  20  44  4E  45  01D00 P.ACU:  .ASCII  \END OF FILE\                          ;
                                                    01D0B         .BLKB   1
                                          0000000B  01D0C P.ACT:  .LONG   11                                     ;
```

```
                              00000000'  01D10           .ADDRESS P.ACU                              ;
                         46  4F  45  01D14 P.ACW:    .ASCII   \EOF\                                  ;
                                     01D17           .BLKB    1
                              00000003  01D18 P.ACV:    .LONG    3                                   ;
                              00000000'  01D1C           .ADDRESS P.ACW                              ;

                                                 SENT132_FORMAT=       P.ACT
                                                 SENT80_FORMAT=        P.ACV


                                  0004 00000 GET_EOF:.WORD    Save R2                                ; 4588
                   52 00000000G  00   9E 00002           MOVAB    SYS$FAO, R2                       ;
                           08    AC   DD 00009           PUSHL    STR_DESC                          ; 4604
                           0C    AC   DD 0000C           PUSHL    RET_LEN                           ;
                           DA    AF   9F 0000F           PUSHAB   SENT132_FORMAT                    ;
                   62             03  FB 00012           CALLS    #3, SYS$FAO                       ;
                   51      0C    BC   3C 00015           MOVZWL   @RET_LEN, R1                      ; 4608
                   51             0C  C4 00019           MULL2    #12, R1                           ;
                   50      04    AC   D0 0001C           MOVL     SCB, R0                           ;
           0200    C0             51  D1 00020           CMPL     R1, 512(R0)                       ;
                                  0C  15 00025           BLEQ     1$                                ;
                           08    AC   DD 00027           PUSHL    STR_DESC                          ; 4613
                           0C    AC   DD 0002A           PUSHL    RET_LEN                           ;
                           C8    AF   9F 0002D           PUSHAB   SENT80_FORMAT                     ;
                   62             03  FB 00030           CALLS    #3, SYS$FAO                       ;
                                      04 00033 1$:       RET                                        ; 4616

; Routine Size:  52 bytes,    Routine Base:  CODE + 1D20
; 3702          4617  1 %sbttl 'GET_ACCOUNTING_INFO - Get the Accounting Information'
; 3703          4618  1 !++
; 3704          4619  1 ! Functional Description:
; 3705          4620  1 !       This routine returns a string containing the accounting information.
; 3706          4621  1 !
; 3707          4622  1 ! Formal Parameters:
; 3708          4623  1 !       SCB             - Address of the SCB
; 3709          4624  1 !       STR_DESC        - Desc of String to Return
; 3710          4625  1 !       RET_LEN         - Return length of Desc.
; 3711          4626  1 !
; 3712          4627  1 ! Implicit Inputs:
; 3713          4628  1 !               none
; 3714          4629  1 !
; 3715          4630  1 ! Implicit Outputs:
; 3716          4631  1 !               none
; 3717          4632  1 !
; 3718          4633  1 ! Returned Value:
; 3719          4634  1 !               none
; 3720          4635  1 !
; 3721          4636  1 ! Side Effects:
; 3722          4637  1 !               none
; 3723          4638  1 !--
; 3724          4639  1 ROUTINE GET_ACCOUNTING_INFO (
; 3725          4640  1               SCB             : REF $BBLOCK,          ! SCB
; 3726          4641  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 3727          4642  1               RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 3728          4643  1                       )       : NOVALUE =
; 3729          4644  2 BEGIN
; 3730          4645  2 BIND
; 3731        P 4646  2     SENT132_FORMAT = $DESCRIPTOR (
; 3732        P 4647  2         '!#(AC)',
; 3733        P 4648  2         '!#(AS)'
; 3734          4649  2                                         );
; 3735          4650  2 LOCAL
; 3736          4651  2     IF_PRES;
; 3737          4652  2
; 3738          4653  2 IF_PRES = .SCB_SIZE_ (ACCOUNTING_DATA);
; 3739          4654  2 IF .IF_PRES GEQ 1
; 3740          4655  2 THEN
; 3741          4656  2     IF_PRES = 1;
; 3742          4657  2
; 3743        P 4658  2 $FAO (  SENT132_FORMAT,
; 3744        P 4659  2         RET_LEN[0],                     ! return length
; 3745        P 4660  2         STR_DESC[0],                    ! address of string
; 3746        P 4661  2         .IF_PRES,
; 3747        P 4662  2         UPLIT BYTE (%ASCIC 'ACCOUNTING INFO:'),
; 3748        P 4663  2         .IF_PRES,
; 3749        P 4664  2         SCB[PSM$Q_ACCOUNTING_DATA]      ! accounting data
; 3750          4665  2     );
; 3751          4666  1 END;


                     29  43  41  28  23  21  01D54 P.ACY:   .ASCII  \!#(AC)\                                  ;
                     29  53  41  28  23  21  01D5A          .ASCII  \!#(AS)\                                  ;
                                  0000000C   01D60 P.ACX:   .LONG   12                                        ;
                                  00000000'  01D64          .ADDRESS P.ACY                                    ;
```

```
46  4E  49  20  47  4E  49  54  4E  55  4F  43  43  41  10  01D68 P.ACZ:   .ASCII  <16>\ACCOUNTING INFO:\                      ;
                                                    3A  4F  01D77                                                               ;

                                                                  SENT132_FORMAT=      P.ACX


                                                       0000 00000 GET_ACCOUNTING_INFO:
                                                                           .WORD   Save nothing                                 ; 4639
                                        50      04  AC  D0 00002           MOVL    SCB, R0                                      ; 4653
                                        51      14  A0  3C 00006           MOVZWL  20(R0), IF_PRES                              ;
                                                    03  15 0000A           BLEQ    1$                                           ; 4654
                                        51          01  D0 0000C           MOVL    #1, IF_PRES                                  ; 4656
                                                14  A0  9F 0000F 1$:       PUSHAB  20(R0)                                       ; 4665
                                                    51  DD 00012           PUSHL   IF_PRES                                      ;
                                                D8  AF  9F 00014           PUSHAB  P.ACZ                                        ;
                                                    51  DD 00017           PUSHL   IF_PRES                                      ;
                                                08  AC  DD 00019           PUSHL   STR_DESC                                     ;
                                                0C  AC  DD 0001C           PUSHL   RET_LEN                                      ;
                                                C5  AF  9F 0001F           PUSHAB  SENT132_FORMAT                               ;
                            00000000G   00          07  FB 00022           CALLS   #7, SYS$FAO                                  ;
                                                        04 00029           RET                                                  ; 4666

; Routine Size:  42 bytes,    Routine Base:  CODE + 1D79
; 3753          4667  1 %sbttl 'GET_QUALIFIERS - Get Switches/Qualifiers associated with PRINTING'
; 3754          4668  1 !++
; 3755          4669  1 ! Functional Description:
; 3756          4670  1 !       This routine returns a string containing the all relevant print
; 3757          4671  1 !       qualifier information.
; 3758          4672  1 !
; 3759          4673  1 ! Formal Parameters:
; 3760          4674  1 !       SCB             - Address of the SCB
; 3761          4675  1 !       STR_DESC        - Desc of String to Return
; 3762          4676  1 !       RET_LEN         - Return length of Desc.
; 3763          4677  1 !
; 3764          4678  1 ! Implicit Inputs:
; 3765          4679  1 !               none
; 3766          4680  1 !
; 3767          4681  1 ! Implicit Outputs:
; 3768          4682  1 !               none
; 3769          4683  1 !
; 3770          4684  1 ! Returned Value:
; 3771          4685  1 !               none
; 3772          4686  1 !
; 3773          4687  1 ! Side Effects:
; 3774          4688  1 !               none
; 3775          4689  1 !--
; 3776          4690  1 ROUTINE GET_QUALIFIERS (
; 3777          4691  1               SCB             : REF $BBLOCK,          ! SCB
; 3778          4692  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 3779          4693  1               RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 3780          4694  1                       )       : NOVALUE =
; 3781          4695  2 BEGIN
; 3782          4696  2 BIND
; 3783          4697  2 ! all the formats start here
; 3784          4698  2 !
; 3785        P 4699  2     DATE_FORMAT = $DESCRIPTOR (
; 3786          4700  2         '!17%D'),
; 3787          4701  2
; 3788        P 4702  2     AFTER_TIME_FORMAT = $DESCRIPTOR (
; 3789          4703  2         ' /AFTER=!17%D'),                             ! - after_time print
; 3790          4704  2
; 3791        P 4705  2     BURST_FORMAT = $DESCRIPTOR (
; 3792          4706  2         ' /BURST'         ),                          ! - burst
; 3793          4707  2
; 3794        P 4708  2     CHARACTERISTICS_FORMAT = $DESCRIPTOR (
; 3795          4709  2         ' /CHARACTERISTICS=!AS'  ),                   ! - characteristics
; 3796          4710  2
; 3797        P 4711  2     FILE_COPIES_FORMAT = $DESCRIPTOR (
; 3798          4712  2         ' /COPIES=!UL(!UL of !UL)'),                  ! - copies/iteration
; 3799          4713  2
; 3800        P 4714  2     FEED_FORMAT = $DESCRIPTOR (
; 3801          4715  2         ' /FEED'                  ),                  ! - feed
; 3802          4716  2
; 3803        P 4717  2     FLAG_FORMAT = $DESCRIPTOR (
; 3804          4718  2         ' /FLAG'                  ),                  ! - flag
; 3805          4719  2
; 3806        P 4720  2     FORM_FORMAT = $DESCRIPTOR (
; 3807          4721  2         ' /FORM=!AS'              ),                  ! - form
; 3808          4722  2
; 3809        P 4723  2     HEADER_FORMAT = $DESCRIPTOR (
; 3810          4724  2            ' /HEADER'                   ),             ! - header
; 3811          4725  2
; 3812        P 4726  2        JOB_COUNT_FORMAT = $DESCRIPTOR (
; 3813          4727  2            ' /JOB_COUNT=!UL(!UL of !UL)'),             ! - job count /iteration
; 3814          4728  2
; 3815        P 4729  2        LENGTH_FORMAT = $DESCRIPTOR(
; 3816          4730  2            ' 7LENGTH=!UL'),
; 3817          4731  2
; 3818        P 4732  2        LIBRARY_FORMAT = $DESCRIPTOR (
; 3819          4733  2            ' /LIBRARY=!AS'  ),                          ! - library
; 3820          4734  2
; 3821        P 4735  2        MARGIN_FORMAT = $DESCRIPTOR(
; 3822          4736  2            ' 7MARGIN=('),
; 3823          4737  2
; 3824        P 4738  2        TOP_FORMAT = $DESCRIPTOR(
; 3825          4739  2            'TOP=!UL'),
; 3826          4740  2
; 3827        P 4741  2        BOTTOM_FORMAT = $DESCRIPTOR(
; 3828          4742  2            'BOTTOM=!UL'),
; 3829          4743  2
; 3830        P 4744  2        LEFT_FORMAT = $DESCRIPTOR(
; 3831          4745  2            'LEFT=!UL'),
; 3832          4746  2
; 3833        P 4747  2        RIGHT_FORMAT = $DESCRIPTOR(
; 3834          4748  2            'RIGHT=!UL'),
; 3835          4749  2
; 3836        P 4750  2        NOFEED_FORMAT = $DESCRIPTOR (
; 3837          4751  2            ' 7NOFEED'                   ),             ! - nofeed
; 3838          4752  2
; 3839        P 4753  2        SETUP_PAGE_FORMAT = $DESCRIPTOR(
; 3840          4754  2            ' /PAGE_SETUP=(!AS)'     ),                 ! - setup page
; 3841          4755  2
; 3842        P 4756  2        PAGES_FORMAT = $DESCRIPTOR (
; 3843          4757  2            ' /PAGES=(!UL,!UL)'      ),                 ! - page count
; 3844          4758  2
; 3845        P 4759  2        PARAMETER_FORMAT = $DESCRIPTOR (
; 3846        P 4760  2            ' /PARAMETERS=(',
; 3847        P 4761  2            '''!AS'', ''!AS'', ''!AS'', ''!AS'', ''!AS'', ',
; 3848          4762  2            '''!AS'', ''!AS'', ''!AS'')'),              ! - parameter lists
; 3849          4763  2
; 3850        P 4764  2        PASSALL_FORMAT = $DESCRIPTOR (
; 3851          4765  2            ' /PASSALL'                  ),             ! - passall
; 3852          4766  2
; 3853        P 4767  2        PUNCTUATION_FORMAT = $DESCRIPTOR(
; 3854          4768  2            '!AC'),                                    ! comma or close paren
; 3855          4769  2
; 3856        P 4770  2        SETUP_FILE_FORMAT = $DESCRIPTOR (
; 3857          4771  2            ' /SETUP_FILE=(!AS)'      ),                ! - setup file
; 3858          4772  2
; 3859        P 4773  2        SETUP_FORM_FORMAT = $DESCRIPTOR(
; 3860          4774  2            ' /SETUP_FORM=(!AS)'     ),                 ! - setup form
; 3861          4775  2
; 3862        P 4776  2        SHEET_FORMAT = $DESCRIPTOR(
; 3863          4777  2            ' /SHEET_FEED'  ),
; 3864          4778  2
; 3865        P 4779  2        SPACE_FORMAT = $DESCRIPTOR (
; 3866          4780  2            ' /SPACE'                    ),             ! - space
; 3867          4781  2
; 3868        P 4782  2       TRAILER_FORMAT = $DESCRIPTOR (
; 3869          4783  2           ' /TRAILER'            ),                  ! - trailer
; 3870          4784  2
; 3871        P 4785  2       TRUNCATE_FORMAT = $DESCRIPTOR(
; 3872          4786  2           ' /TRUNCATE'     ),
; 3873          4787  2
; 3874        P 4788  2       WIDTH_FORMAT = $DESCRIPTOR(
; 3875          4789  2           '/WIDTH=!UL'),
; 3876          4790  2
; 3877        P 4791  2       WRAP_FORMAT = $DESCRIPTOR(
; 3878          4792  2           ' /WRAP'          );
; 3879          4793  2
; 3880          4794  2 LITERAL
; 3881          4795  2     K_MAX_BUFFER_SIZE = 512;
; 3882          4796  2
; 3883          4797  2 LOCAL
; 3884          4798  2     PUNC_FLAG    : INITIAL (0),
; 3885          4799  2     TEMP_LEN     ,
; 3886          4800  2     AFT_DATE_PTR: VECTOR[2],
; 3887          4801  2     TEMP_PTR     : VECTOR[2],
; 3888          4802  2     AFT_BUFF     : VECTOR[17,byte],
; 3889          4803  2     TEMP_BUFF    : VECTOR[17,byte],
; 3890          4804  2     IF_PRES      ,
; 3891          4805  2     CURRENT_LEN : INITIAL (0),
; 3892          4806  2     STRING_PTR  : VECTOR [2];              ! Pointer to current string
; 3893          4807  2
; 3894          4808  2 ! Allocate the buffer for "GET_xxx" Routines
; 3895          4809  2 !
; 3896          4810  2 STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE;
; 3897          4811  2 STRING_PTR[ADDR] = .STR_DESC[ADDR];      ! init address
; 3898          4812  2
; 3899          4813  2 RET_LEN[0] = 0;
; 3900          4814  2
; 3901          4815  2 !$FAO (  BEGIN_FORMAT,
; 3902          4816  2 !        CURRENT_LEN,                        ! return length
; 3903          4817  2 !        STRING_PTR[0],                      ! address of string
; 3904          4818  2  !     );
; 3905          4819  2 !
; 3906          4820  2 !RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 3907          4821  2 !STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 3908          4822  2 !STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 3909          4823  2
; 3910          4824  2
; 3911          4825  2 ! Increment pointer only if not equal to time queued
; 3912          4826  2 !
; 3913          4827  2 AFT_DATE_PTR[SIZE] = %ALLOCATION(AFT_BUFF);
; 3914          4828  2 AFT_DATE_PTR[ADDR] = AFT_BUFF;
; 3915          4829  2
; 3916        P 4830  2 $FAO (  DATE_FORMAT,
; 3917        P 4831  2         TEMP_LEN,
; 3918        P 4832  2         AFT_DATE_PTR[0],
; 3919          4833  2         SCB[PSM$@_AFTER_TIME]);
; 3920          4834  2
; 3921          4835  2 TEMP_PTR[SIZE] = %ALLOCATION(TEMP_BUFF);
; 3922          4836  2 TEMP_PTR[ADDR] = TEMP_BUFF;
; 3923          4837  2
; 3924     P 4838  2 $FAO (  DATE_FORMAT,
; 3925     P 4839  2         TEMP_LEN,
; 3926     P 4840  2         TEMP_PTR[0],
; 3927       4841  2         SCB[PSM$Q_TIME_QUEUED]);
; 3928       4842  2
; 3929       4843  2 IF CH$NEQ( .TEMP_LEN, .TEMP_PTR[ADDR], .TEMP_LEN, .AFT_DATE_PTR[ADDR])
; 3930       4844  2 THEN
; 3931       4845  3     BEGIN
; 3932       4846  3
; 3933     P 4847  3     $FAO (
; 3934     P 4848  3         AFTER_TIME_FORMAT,
; 3935     P 4849  3         CURRENT_LEN,                      ! return length
; 3936     P 4850  3         STRING_PTR[0],                    ! address of string
; 3937       4851  3         SCB[PSM$Q_AFTER_TIME]);           ! after_time
; 3938       4852  3
; 3939       4853  3     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 3940       4854  3     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 3941       4855  3     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 3942       4856  2     END;
; 3943       4857  2
; 3944       4858  2 IF .SEPARATE_FLAG_ (FILE_BURST)
; 3945       4859  2 THEN
; 3946       4860  3     BEGIN
; 3947     P 4861  3     $FAO (  BURST_FORMAT,
; 3948     P 4862  3             CURRENT_LEN,                          ! return length
; 3949     P 4863  3             STRING_PTR[0]                         ! address of string
; 3950       4864  3         );
; 3951       4865  3
; 3952       4866  3     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 3953       4867  3     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 3954       4868  3     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 3955       4869  2     END;
; 3956       4870  2
; 3957       4871  3 IF (.SCB_SIZE_ (CHARACTERISTICS) EQL 0)
; 3958       4872  2 THEN
; 3959       4873  3     BEGIN
; 3960     P 4874  3     $FAO (  CHARACTERISTICS_FORMAT,
; 3961     P 4875  3             CURRENT_LEN,                          ! return length
; 3962     P 4876  3             STRING_PTR[0],                        ! address of string
; 3963     P 4877  3             SCB[PSM$Q_CHARACTERISTICS]            ! /CHARACTERISTICS
; 3964       4878  3         );
; 3965       4879  3
; 3966       4880  3     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 3967       4881  3     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 3968       4882  3     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 3969       4883  2     END;
; 3970       4884  2
; 3971       4885  2 ! Always print something about form feed... /FEED or /NOFEED
; 3972       4886  2 !
; 3973       4887  2 IF .$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_PAGINATE]
; 3974       4888  2 THEN
; 3975       4889  3     BEGIN
; 3976     P 4890  3     $FAO (  FEED_FORMAT,
; 3977     P 4891  3             CURRENT_LEN,                          ! return length
; 3978     P 4892  3             STRING_PTR[0]                         ! address of string
; 3979       4893  3         );
; 3980       4894  3
; 3981          4895  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 3982          4896  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 3983          4897  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 3984          4898  2         END;
; 3985          4899  2
; 3986          4900  2     IF .SCB[PSM$L_FILE_COPIES] GTR 1
; 3987          4901  2     THEN
; 3988          4902  3         BEGIN
; 3989        P 4903  3         $FAO (  FILE_COPIES_FORMAT,
; 3990        P 4904  3                 CURRENT_LEN,                            ! return length
; 3991        P 4905  3                 STRING_PTR[0],                          ! address of string
; 3992        P 4906  3                 .SCB[PSM$L_FILE_COPIES],                ! /FILE_COPIES
; 3993        P 4907  3                 .SCB[PSM$L_FILE_COUNT],                 ! iteration
; 3994        P 4908  3                 .SCB[PSM$L_FILE_COPIES]
; 3995          4909  3             );
; 3996          4910  3
; 3997          4911  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 3998          4912  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 3999          4913  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4000          4914  2         END;
; 4001          4915  2
; 4002          4916  2     IF .SEPARATE_FLAG_ (FILE_FLAG)
; 4003          4917  2     THEN
; 4004          4918  3         BEGIN
; 4005        P 4919  3         $FAO (  FLAG_FORMAT,
; 4006        P 4920  3                 CURRENT_LEN,                            ! return length
; 4007        P 4921  3                 STRING_PTR[0]                           ! address of string
; 4008          4922  3             );
; 4009          4923  3
; 4010          4924  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4011          4925  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4012          4926  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4013          4927  2         END;
; 4014          4928  2
; 4015        P 4929  2     $FAO (  FORM_FORMAT,
; 4016        P 4930  2             CURRENT_LEN,                        ! return length
; 4017        P 4931  2             STRING_PTR[0],                      ! address of string
; 4018          4932  2             SCB[PSM$Q_FORM_NAME]);
; 4019          4933  2
; 4020          4934  2     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4021          4935  2     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4022          4936  2     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4023          4937  2
; 4024          4938  2     IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4025          4939  2     THEN
; 4026          4940  3         BEGIN
; 4027          4941  3             RET_LEN[0] = 512;
; 4028          4942  3             RETURN;
; 4029          4943  2         END;
; 4030          4944  2
; 4031          4945  2     IF .$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_PAGE_HEADER]
; 4032          4946  2     THEN
; 4033          4947  3         BEGIN
; 4034        P 4948  3         $FAO (  HEADER_FORMAT,
; 4035        P 4949  3                 CURRENT_LEN,                            ! return length
; 4036        P 4950  3                 STRING_PTR[0]                           ! address of string
; 4037          4951  3             );
```

```
   3
   3        RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
   3        STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
   3        STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
   2        END;
   2
   2  IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
   2  THEN
   3        BEGIN
   3            RET_LEN[0] = 512;
   3            RETURN;
   2        END;
   2
   2  IF .SCB[PSM$L_JOB_COPIES] GTR 1
   2  THEN
   3        BEGIN
P  3        $FAO (  JOB_COUNT_FORMAT,
P  3                CURRENT_LEN,                            ! return length
P  3                STRING_PTR[0],                          ! address of string
P  3                .SCB[PSM$L_JOB_COPIES],                 ! /JOB_COUNT
P  3                .SCB[PSM$L_JOB_COUNT],                  ! iteration
P  3                .SCB[PSM$L_JOB_COPIES]                  !
   3            );
   3
   3        RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
   3        STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
   3        STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
   2        END;
   2
   2  ! Here is my internal call to PSM$READ_ITEM_DX to insure that the user
   2  ! can copy information successfully using this routine
   2  !
   3  BEGIN
   3  LOCAL LEN: VECTOR[2];
   3
   3  INIT_DYN_DESC_ (LEN);
   3  PSM$READ_ITEM_DX (.SCB , %ref(SMBMSG$K_FORM_LENGTH),
   3                          LEN[0]);
   3
P  3        $FAO (  LENGTH_FORMAT,
P  3                CURRENT_LEN,                            ! return length
P  3                STRING_PTR[0],                          ! address of string
P  3                ..LEN[ADDR]                             ! length pointed to by len[addr]
   3            );
   3
   3  RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
   3  STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
   3  STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
   2  END;
   2
   2  !***  ALWAYS PRINT THE LIBRARY !**!
   2  !
P  2  $FAO (  LIBRARY_FORMAT,
P  2          CURRENT_LEN,                    ! return length
P  2          STRING_PTR[0],                  ! address of string
P  2          SCB[PSM$Q_LIBRARY_SPECIFICATION]! /LIBRARY
   2      );
```

```
; 4095        5009  2
; 4096        5010  2 RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4097        5011  2 STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4098        5012  2 STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4099        5013  2
; 4100        5014  2 IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4101        5015  2 THEN
; 4102        5016  3     BEGIN
; 4103        5017  3         RET_LEN[0] = 512;
; 4104        5018  3         RETURN;
; 4105        5019  2     END;
; 4106        5020  2
; 4107        5021  2 IF (.SCB[PSM$L_TOP_MARGIN] NEQ 0) OR
; 4108        5022  2     (.SCB[PSM$L_BOTTOM_MARGIN] NEQ 0) OR
; 4109        5023  2     (.SCB[PSM$L_LEFT_MARGIN] NEQ 0) OR
; 4110        5024  3     (.SCB[PSM$L_RIGHT_MARGIN] NEQ 0)
; 4111        5025  2 THEN
; 4112        5026  3     BEGIN
; 4113        5027  3
; 4114      P 5028  3     $FAO (  MARGIN_FORMAT,
; 4115      P 5029  3             CURRENT_LEN,                             ! return length
; 4116      P 5030  3             STRING_PTR[0]                            ! address of string
; 4117        5031  3         );
; 4118        5032  3
; 4119        5033  3     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4120        5034  3     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4121        5035  3     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4122        5036  3
; 4123        5037  4     IF (.SCB[PSM$L_TOP_MARGIN] NEQ 0)
; 4124        5038  3     THEN
; 4125        5039  4         BEGIN
; 4126      P 5040  4         $FAO (  TOP_FORMAT,
; 4127      P 5041  4                 CURRENT_LEN,                         ! return length
; 4128      P 5042  4                 STRING_PTR[0],                       ! address of string
; 4129      P 5043  4                 .SCB[PSM$L_TOP_MARGIN]               ! top
; 4130        5044  4             );
; 4131        5045  4
; 4132        5046  4         PUNC_FLAG = 1;
; 4133        5047  4         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4134        5048  4         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4135        5049  4         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4136        5050  3         END;
; 4137        5051  3
; 4138        5052  3         IF  (.SCB[PSM$L_BOTTOM_MARGIN] NEQ 0) AND
; 4139        5053  3             .PUNC_FLAG
; 4140        5054  3         THEN
; 4141        5055  4             BEGIN
; 4142      P 5056  4             $FAO(
; 4143      P 5057  4                 PUNCTUATION_FORMAT,
; 4144      P 5058  4                 CURRENT_LEN,                         ! return length
; 4145      P 5059  4                 STRING_PTR[0],                       ! address of string
; 4146      P 5060  4                 UPLIT BYTE (%ASCIC ',')
; 4147        5061  4                 );
; 4148        5062  4
; 4149        5063  4             RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4150        5064  4             STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4151        5065  4             STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4152        5066 3              END;
; 4153        5067 3
; 4154        5068 4          IF  (.SCB[PSM$L_BOTTOM_MARGIN] NEQ 0)
; 4155        5069 3          THEN
; 4156        5070 4              BEGIN
; 4157      P 5071 4              $FAO (
; 4158      P 5072 4                  BOTTOM_FORMAT,
; 4159      P 5073 4                  CURRENT_LEN,                        ! return length
; 4160      P 5074 4                  STRING_PTR[0],                      ! address of string
; 4161      P 5075 4                  .SCB[PSM$L_BOTTOM_MARGIN]           ! bottom
; 4162        5076 4                  );
; 4163        5077 4
; 4164        5078 4              PUNC_FLAG = 1;
; 4165        5079 4              RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4166        5080 4              STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4167        5081 4              STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4168        5082 3              END;
; 4169        5083 3
; 4170        5084 3          IF  (.SCB[PSM$L_LEFT_MARGIN] NEQ 0) AND
; 4171        5085 3              .PUNC_FLAG
; 4172        5086 3          THEN
; 4173        5087 4              BEGIN
; 4174      P 5088 4              $FAO(
; 4175      P 5089 4                  PUNCTUATION_FORMAT,
; 4176      P 5090 4                  CURRENT_LEN,                        ! return length
; 4177      P 5091 4                  STRING_PTR[0],                      ! address of string
; 4178      P 5092 4                  UPLIT BYTE (%ASCIC ',')
; 4179        5093 4                  );
; 4180        5094 4              RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4181        5095 4              STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4182        5096 4              STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4183        5097 3          END;
; 4184        5098 3
; 4185        5099 4          IF  (.SCB[PSM$L_LEFT_MARGIN] NEQ 0)
; 4186        5100 3          THEN
; 4187        5101 4              BEGIN
; 4188      P 5102 4              $FAO (  LEFT_FORMAT,
; 4189      P 5103 4                  CURRENT_LEN,                        ! return length
; 4190      P 5104 4                  STRING_PTR[0],                      ! address of string
; 4191      P 5105 4                  .SCB[PSM$L_LEFT_MARGIN]             ! left
; 4192        5106 4                  );
; 4193        5107 4
; 4194        5108 4              PUNC_FLAG = 1;
; 4195        5109 4              RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4196        5110 4              STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4197        5111 4              STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4198        5112 3              END;
; 4199        5113 3
; 4200        5114 3          IF  (.SCB[PSM$L_RIGHT_MARGIN] NEQ 0) AND
; 4201        5115 3              .PUNC_FLAG
; 4202        5116 3          THEN
; 4203        5117 4              BEGIN
; 4204      P 5118 4              $FAO(
; 4205      P 5119 4                  PUNCTUATION_FORMAT,
; 4206      P 5120 4                  CURRENT_LEN,                        ! return length
; 4207      P 5121 4                  STRING_PTR[0],                      ! address of string
; 4208      P 5122 4                  UPLIT BYTE (%ASCIC ',')
; 4209          5123  4                             );
; 4210          5124  4
; 4211          5125  4                     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4212          5126  4                     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4213          5127  4                     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4214          5128  3                     END;
; 4215          5129  3
; 4216          5130  4                 IF   (.SCB[PSM$L_RIGHT_MARGIN] NEQ 0)
; 4217          5131  3                 THEN
; 4218          5132  4                 BEGIN
; 4219        P 5133  4                     $FAO (  RIGHT_FORMAT,
; 4220        P 5134  4                             CURRENT_LEN,                    ! return length
; 4221        P 5135  4                             STRING_PTR[0],                  ! address of string
; 4222        P 5136  4                             .SCB[PSM$L_RIGHT_MARGIN],       ! right
; 4223          5137  4                         );
; 4224          5138  4
; 4225          5139  4                     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4226          5140  4                     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4227          5141  4                     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4228          5142  3                 END;
; 4229          5143  3
; 4230        P 5144  3                 $FAO(
; 4231        P 5145  3                     PUNCTUATION_FORMAT,
; 4232        P 5146  3                     CURRENT_LEN,                            ! return length
; 4233        P 5147  3                     STRING_PTR[0],                          ! address of string
; 4234        P 5148  3                     UPLIT BYTE (%ASCIC ')')
; 4235          5149  3                     );
; 4236          5150  3
; 4237          5151  3                 RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4238          5152  3                 STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4239          5153  3                 STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4240          5154  2             END;
; 4241          5155  2
; 4242          5156  2     IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4243          5157  2     THEN
; 4244          5158  3             BEGIN
; 4245          5159  3                 RET_LEN[0] = 512;
; 4246          5160  3                 RETURN;
; 4247          5161  2             END;
; 4248          5162  2
; 4249          5163  3     IF NOT (.$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_PAGINATE])
; 4250          5164  2     THEN
; 4251          5165  3             BEGIN
; 4252        P 5166  3             $FAO (  NOFEED_FORMAT,
; 4253        P 5167  3                     CURRENT_LEN,                            ! return length
; 4254        P 5168  3                     STRING_PTR[0]                           ! address of string
; 4255          5169  3                 );
; 4256          5170  3
; 4257          5171  3             RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4258          5172  3             STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4259          5173  3             STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4260          5174  2             END;
; 4261          5175  2
; 4262          5176  2     IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4263          5177  2     THEN
; 4264          5178  3             BEGIN
; 4265          5179  3                 RET_LEN[0] = 512;
; 4266           5180  3            RETURN;
; 4267           5181  2        END;
; 4268           5182  2
; 4269           5183  2  IF (.SCB[PSM$L_FIRST_PAGE] NEQ 0) OR                    ! default last page is zero
; 4270           5184  3      (.SCB[PSM$[_LAST_PAGE] NEQ 0)
; 4271           5185  2  THEN
; 4272           5186  3      BEGIN
; 4273         P 5187  3      $FAO (  PAGES_FORMAT,
; 4274         P 5188  3              CURRENT_LEN,                                ! return length
; 4275         P 5189  3                  STRING_PTR[0],                          ! address of string
; 4276         P 5190  3                  .SCB[PSM$L_FIRST_PAGE],
; 4277         P 5191  3                  .SCB[PSM$L_LAST_PAGE]
; 4278           5192  3            );
; 4279           5193  3
; 4280           5194  3      RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4281           5195  3      STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4282           5196  3      STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4283           5197  2      END;
; 4284           5198  2
; 4285           5199  2  IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4286           5200  2  THEN
; 4287           5201  3      BEGIN
; 4288           5202  3          RET_LEN[0] = 512;
; 4289           5203  3          RETURN;
; 4290           5204  2      END;
; 4291           5205  2
; 4292           5206  3  IF (.SCB_SIZE_ (PARAMETER_1) NEQ 0 OR
; 4293           5207  3      .SCB_SIZE_ (PARAMETER_2) NEQ 0 OR
; 4294           5208  3      .SCB_SIZE_ (PARAMETER_3) NEQ 0 OR
; 4295           5209  3      .SCB_SIZE_ (PARAMETER_4) NEQ 0 OR
; 4296           5210  3      .SCB_SIZE_ (PARAMETER_5) NEQ 0 OR
; 4297           5211  3      .SCB_SIZE_ (PARAMETER_6) NEQ 0 OR
; 4298           5212  3      .SCB_SIZE_ (PARAMETER_7) NEQ 0 OR
; 4299           5213  3      .SCB_SIZE_ (PARAMETER_8) NEQ 0  )
; 4300           5214  2  THEN
; 4301           5215  3      BEGIN
; 4302         P 5216  3      $FAO (
; 4303         P 5217  3          PARAMETER_FORMAT,
; 4304         P 5218  3          CURRENT_LEN,                       ! return length
; 4305         P 5219  3          STRING_PTR[0],                     ! address of string
; 4306         P 5220  3          SCB[PSM$Q_PARAMETER_1],            ! P1
; 4307         P 5221  3          SCB[PSM$Q_PARAMETER_2],            ! P2
; 4308         P 5222  3          SCB[PSM$Q_PARAMETER_3],            ! P3
; 4309         P 5223  3          SCB[PSM$Q_PARAMETER_4],            ! P4
; 4310         P 5224  3          SCB[PSM$Q_PARAMETER_5],            ! P5
; 4311         P 5225  3          SCB[PSM$Q_PARAMETER_6],            ! P6
; 4312         P 5226  3          SCB[PSM$Q_PARAMETER_7],            ! P7
; 4313         P 5227  3          SCB[PSM$Q_PARAMETER_8]             ! P8
; 4314           5228  3       );
; 4315           5229  3
; 4316           5230  3      RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4317           5231  3      STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4318           5232  3      STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4319           5233  2      END;
; 4320           5234  2
; 4321           5235  2  IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4322           5236  2  THEN
; 4323          5237  3         BEGIN
; 4324          5238  3             RET_LEN[0] = 512;
; 4325          5239  3             RETURN;
; 4326          5240  2         END;
; 4327          5241  2
; 4328          5242  2     IF .$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_PASSALL]
; 4329          5243  2     THEN
; 4330          5244  3         BEGIN
; 4331        P 5245  3         $FAO (  PASSALL_FORMAT,
; 4332        P 5246  3                 CURRENT_LEN,                            ! return length
; 4333        P 5247  3                 STRING_PTR[0]                           ! address of string
; 4334          5248  3             );
; 4335          5249  3
; 4336          5250  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4337          5251  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4338          5252  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4339          5253  2         END;
; 4340          5254  2
; 4341          5255  2     IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4342          5256  2     THEN
; 4343          5257  3         BEGIN
; 4344          5258  3             RET_LEN[0] = 512;
; 4345          5259  3             RETURN;
; 4346          5260  2         END;
; 4347          5261  2
; 4348          5262  2     IF .SCB_SIZE_ (FILE_SETUP_MODULES) GTR 0
; 4349          5263  2     THEN
; 4350          5264  3         BEGIN
; 4351        P 5265  3         $FAO (  SETUP_FILE_FORMAT,
; 4352        P 5266  3                 CURRENT_LEN,                            ! return length
; 4353        P 5267  3                 STRING_PTR[0],                          ! address of string
; 4354        P 5268  3                 SCB[PSM$Q_FILE_SETUP_MODULES]
; 4355          5269  3             );
; 4356          5270  3
; 4357          5271  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4358          5272  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4359          5273  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4360          5274  2         END;
; 4361          5275  2
; 4362          5276  2     IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4363          5277  2     THEN
; 4364          5278  3         BEGIN
; 4365          5279  3             RET_LEN[0] = 512;
; 4366          5280  3             RETURN;
; 4367          5281  2         END;
; 4368          5282  2
; 4369          5283  2     IF .SCB_SIZE_ (FORM_SETUP_MODULES) GTR 0
; 4370          5284  2     THEN
; 4371          5285  3         BEGIN
; 4372        P 5286  3         $FAO (  SETUP_FORM_FORMAT,
; 4373        P 5287  3                 CURRENT_LEN,                            ! return length
; 4374        P 5288  3                 STRING_PTR[0],                          ! address of string
; 4375        P 5289  3                 SCB[PSM$Q_FORM_SETUP_MODULES]           ! form setup
; 4376          5290  3             );
; 4377          5291  3
; 4378          5292  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4379          5293  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
```

```
  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
  2         END;
  2
  2    IF .SCB_SIZE_ (PAGE_SETUP_MODULES) GTR 0
  2    THEN
  3         BEGIN
P 3         $FAO (  SETUP_PAGE_FORMAT,
P 3                 CURRENT_LEN,                          ! return length
P 3                 STRING_PTR[0],                        ! address of string
P 3                 SCB[PSM$Q_PAGE_SETUP_MODULES]         ! form setup
  3              );
  3
  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
  2         END;
  2
  2    IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
  2    THEN
  3         BEGIN
  3              RET_LEN[0] = 512;
  3              RETURN;
  2         END;
  2
  2    IF .$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_SHEET_FEED]
  2    THEN
  3         BEGIN
P 3         $FAO (  SHEET_FORMAT,
P 3                 CURRENT_LEN,                          ! return length
P 3                 STRING_PTR[0]                         ! address of string
  3              );
  3
  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
  2         END;
  2
  2    IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
  2    THEN
  3         BEGIN
  3              RET_LEN[0] = 512;
  3              RETURN;
  2         END;
  2
  2    IF .$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_DOUBLE_SPACE]
  2    THEN
  3         BEGIN
P 3         $FAO (  SPACE_FORMAT,
P 3                 CURRENT_LEN,                          ! return length
P 3                 STRING_PTR[0]                         ! address of string
  3              );
  3
  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
  2         END;
  2
```

```
; 4437          5351  2 IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4438          5352  2 THEN
; 4439          5353  3     BEGIN
; 4440          5354  3         RET_LEN[0] = 512;
; 4441          5355  3         RETURN;
; 4442          5356  2     END;
; 4443          5357  2
; 4444          5358  2 IF .SEPARATE_FLAG_ (FILE_TRAILER)
; 4445          5359  2 THEN
; 4446          5360  3     BEGIN
; 4447        P 5361  3     $FAO (  TRAILER_FORMAT,
; 4448        P 5362  3             CURRENT_LEN,                            ! return length
; 4449        P 5363  3             STRING_PTR[0]                           ! address of string
; 4450          5364  3         );
; 4451          5365  3
; 4452          5366  3     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4453          5367  3     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4454          5368  3     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4455          5369  2     END;
; 4456          5370  2
; 4457          5371  2 IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4458          5372  2 THEN
; 4459          5373  3     BEGIN
; 4460          5374  3         RET_LEN[0] = 512;
; 4461          5375  3         RETURN;
; 4462          5376  2     END;
; 4463          5377  2
; 4464          5378  2 IF .$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_TRUNCATE]
; 4465          5379  2 THEN
; 4466          5380  3     BEGIN
; 4467        P 5381  3     $FAO (  TRUNCATE_FORMAT,
; 4468        P 5382  3             CURRENT_LEN,                            ! return length
; 4469        P 5383  3             STRING_PTR[0]                           ! address of string
; 4470          5384  3         );
; 4471          5385  3
; 4472          5386  3     RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4473          5387  3     STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4474          5388  3     STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4475          5389  2     END;
; 4476          5390  2
; 4477          5391  2 IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4478          5392  2 THEN
; 4479          5393  3     BEGIN
; 4480          5394  3         RET_LEN[0] = 512;
; 4481          5395  3         RETURN;
; 4482          5396  2     END;
; 4483          5397  2
; 4484        P 5398  2 $FAO (  WIDTH_FORMAT,
; 4485        P 5399  2         CURRENT_LEN,                        ! return length
; 4486        P 5400  2         STRING_PTR[0],                      ! address of string
; 4487        P 5401  2         .SCB[PSM$L_FORM_WIDTH]              ! form width
; 4488          5402  2     );
; 4489          5403  2
; 4490          5404  2 RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4491          5405  2 STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4492          5406  2 STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4493          5407  2
; 4494          5408  2 IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4495          5409  2 THEN
; 4496          5410  3       BEGIN
; 4497          5411  3           RET_LEN[0] = 512;
; 4498          5412  3           RETURN;
; 4499          5413  2       END;
; 4500          5414  2
; 4501          5415  2 IF .$BBLOCK[SCB[PSM$L_PRINT_CONTROL], SMBMSG$V_WRAP]
; 4502          5416  2 THEN
; 4503          5417  3       BEGIN
; 4504        P 5418  3       $FAO (  WRAP_FORMAT,
; 4505        P 5419  3               CURRENT_LEN,                            ! return length
; 4506        P 5420  3               STRING_PTR[0]                           ! address of string
; 4507          5421  3           );
; 4508          5422  3
; 4509          5423  3       RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4510          5424  2       END;
; 4511          5425  2
; 4512          5426  2 ! Don't print anything if no flags were set
; 4513          5427  2 !
; 4514          5428  2 IF .RET_LEN[0] LEQ 18
; 4515          5429  2 THEN
; 4516          5430  2       RET_LEN[0] = 0;
; 4517          5431  2
; 4518          5432  2 ! Length returned must be less than max string size
; 4519          5433  2 !
; 4520          5434  2 IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4521          5435  2 THEN
; 4522          5436  3       BEGIN
; 4523          5437  3           RET_LEN[0] = 512;
; 4524          5438  3           RETURN;
; 4525          5439  2       END;
; 4526          5440  2
; 4527          5441  1 END;


                                   44  25  37  31  21  01DA3 P.ADB:  .ASCII  \!17%D\                               ;
                                            00000005.  01DA8 P.ADA:  .LONG   5                                     ;
                                            00000000'  01DAC         .ADDRESS P.ADB                                ;
      44  25  37  31  21  3D  52  45  54  46  41  2F  20  01DB0 P.ADD:  .ASCII  \ /AFTER=!17%D\                    ;
                                                       01DBD         .BLKB   3
                                            0000000D   01DC0 P.ADC:  .LONG   13                                    ;
                                            00000000'  01DC4         .ADDRESS P.ADD                                ;
                              54  53  52  55  42  2F  20  01DC8 P.ADF:  .ASCII  \ /BURST\                          ;
                                                       01DCF         .BLKB   1
                                            00000007   01DD0 P.ADE:  .LONG   7                                     ;
                                            00000000'  01DD4         .ADDRESS P.ADF                                ;
49  54  53  49  52  45  54  43  41  52  41  48  43  2F  20  01DD8 P.ADH:  .ASCII  \ /CHARACTERISTICS=!AS\          ;
                                    53  41  21  3D  53  43  01DE7                                                  ;
                                                       01DED         .BLKB   3
                                            00000015   01DF0 P.ADG:  .LONG   21                                    ;
                                            00000000'  01DF4         .ADDRESS P.ADH                                ;
55  21  28  4C  55  21  3D  53  45  49  50  4F  43  2F  20  01DF8 P.ADJ:  .ASCII  \ /COPIES=!UL(!UL of !UL)\       ;
                        29  4C  55  21  20  66  6F  20  4C  01E07                                                  ;
                                            00000018   01E10 P.ADI:  .LONG   24                                    ;
                                            00000000'  01E14         .ADDRESS P.ADJ                                ;
```

```
                           44 45 45 46 2F 20  01E18 P.ADL:  .ASCII  \ /FEED\                        ;
                                              01E1E         .BLKB   2
                                    00000006  01E20 P.ADK:  .LONG   6                               ;
                                    00000000' 01E24         .ADDRESS P.ADL                          ;
                           47 41 4C 46 2F 20  01E28 P.ADN:  .ASCII  \ /FLAG\                        ;
                                              01E2E         .BLKB   2
                                    00000006  01E30 P.ADM:  .LONG   6                               ;
                                    00000000' 01E34         .ADDRESS P.ADN                          ;
               53 41 21 3D 4D 52 4F 46 2F 20  01E38 P.ADP:  .ASCII  \ /FORM=!AS\                    ;
                                              01E42         .BLKB   2
                                    0000000A  01E44 P.ADO:  .LONG   10                              ;
                                    00000000' 01E48         .ADDRESS P.ADP                          ;
                     52 45 44 41 45 48 2F 20  01E4C P.ADR:  .ASCII  \ /HEADER\                      ;
                                    00000008  01E54 P.ADQ:  .LONG   8                               ;
                                    00000000' 01E58         .ADDRESS P.ADR                          ;
4C 55 21 3D 54 4E 55 4F 43 5F 42 4F 4A 2F 20  01E5C P.ADT:  .ASCII  \ /JOB_COUNT=!UL(!UL of !UL)\   ;
         29 4C 55 21 20 66 6F 20 4C 55 21 28  01E6B                                                 ;
                                              01E77         .BLKB   1
                                    0000001B  01E78 P.ADS:  .LONG   27                              ;
                                    00000000' 01E7C         .ADDRESS P.ADT                          ;
         4C 55 21 3D 48 54 47 4E 45 4C 2F 20  01E80 P.ADV:  .ASCII  \ /LENGTH=!UL\                  ;
                                    0000000C  01E8C P.ADU:  .LONG   12                              ;
                                    00000000' 01E90         .ADDRESS P.ADV                          ;
      53 41 21 3D 59 52 41 52 42 49 4C 2F 20  01E94 P.ADX:  .ASCII  \ /LIBRARY=!AS\                 ;
                                              01EA1         .BLKB   3
                                    0000000D  01EA4 P.ADW:  .LONG   13                              ;
                                    00000000' 01EA8         .ADDRESS P.ADX                          ;
               28 3D 4E 49 47 52 41 4D 2F 20  01EAC P.ADZ:  .ASCII  \ /MARGIN=(\                    ;
                                              01EB6         .BLKB   2
                                    0000000A  01EB8 P.ADY:  .LONG   10                              ;
                                    00000000' 01EBC         .ADDRESS P.ADZ                          ;
                        4C 55 21 3D 50 4F 54  01EC0 P.AEB:  .ASCII  \TOP=!UL\                       ;
                                              01EC7         .BLKB   1
                                    00000007  01EC8 P.AEA:  .LONG   7                               ;
                                    00000000' 01ECC         .ADDRESS P.AEB                          ;
               4C 55 21 3D 4D 4F 54 54 4F 42  01ED0 P.AED:  .ASCII  \BOTTOM=!UL\                    ;
                                              01EDA         .BLKB   2
                                    0000000A  01EDC P.AEC:  .LONG   10                              ;
                                    00000000' 01EE0         .ADDRESS P.AED                          ;
                     4C 55 21 3D 54 46 45 4C  01EE4 P.AEF:  .ASCII  \LEFT=!UL\                      ;
                                    00000008  01EEC P.AEE:  .LONG   8                               ;
                                    00000000' 01EF0         .ADDRESS P.AEF                          ;
                  4C 55 21 3D 54 48 47 49 52  01EF4 P.AEH:  .ASCII  \RIGHT=!UL\                     ;
                                              01EFD         .BLKB   3
                                    00000009  01F00 P.AEG:  .LONG   9                               ;
                                    00000000' 01F04         .ADDRESS P.AEH                          ;
                     44 45 45 46 4F 4E 2F 20  01F08 P.AEJ:  .ASCII  \ /NOFEED\                      ;
                                    00000008  01F10 P.AEI:  .LONG   8                               ;
                                    00000000' 01F14         .ADDRESS P.AEJ                          ;
21 28 3D 50 55 54 45 53 5F 45 47 41 50 2F 20  01F18 P.AEL:  .ASCII  \ /PAGE_SETUP=(!AS)\            ;
                                    29 53 41  01F27                                                 ;
                                              01F2A         .BLKB   2
                                    00000012  01F2C P.AEK:  .LONG   18                              ;
                                    00000000' 01F30         .ADDRESS P.AEL                          ;
55 21 2C 4C 55 21 28 3D 53 45 47 41 50 2F 20  01F34 P.AEN:  .ASCII  \ /PAGES=(!UL,!UL)\             ;
                                       29 4C  01F43                                                 ;
                                              01F45         .BLKB   3
```

```
                                                  00000011  01F48 P.AEM:  .LONG    17                                                     ;
                                                  00000000' 01F4C         .ADDRESS P.AEN                                                  ;
    28  3D  53  52  45  54  45  4D  41  52  41  50  2F  20  01F50 P.AEP:  .ASCII   \ /PARAMETERS=(\                                       ;
22  20  2C  22  53  41  21  22  20  2C  22  53  41  21  22  01F5E         .ASCII   \"!AS", "!AS", "!AS", "!AS", "!AS", \                 ;
21  22  20  2C  22  53  41  21  22  20  2C  22  53  41  21  01F6D                                                                         ;
                                        20  2C  22  53  41  01F7C                                                                         ;
22  20  2C  22  53  41  21  22  20  2C  22  53  41  21  22  01F81         .ASCII   \"!AS", "!AS", "!AS")\                                 ;
                                        29  22  53  41  21  01F90                                                                         ;
                                                            01F95         .BLKB    3
                                                  00000045  01F98 P.AEO:  .LONG    69                                                     ;
                                                  00000000' 01F9C         .ADDRESS P.AEP                                                  ;
                        4C  4C  41  53  53  41  50  2F  20  01FA0 P.AER:  .ASCII   \ /PASSALL\                                            ;
                                                            01FA9         .BLKB    3
                                                  00000009  01FAC P.AEQ:  .LONG    9                                                      ;
                                                  00000000' 01FB0         .ADDRESS P.AER                                                  ;
                                                43  41  21  01FB4 P.AET:  .ASCII   \!AC\                                                  ;
                                                            01FB7         .BLKB    1
                                                  00000003  01FB8 P.AES:  .LONG    3                                                      ;
                                                  00000000' 01FBC         .ADDRESS P.AET                                                  ;
21  28  3D  45  4C  49  46  5F  50  55  54  45  53  2F  20  01FC0 P.AEV:  .ASCII   \ /SETUP_FILE=(!AS)\                                   ;
                                                29  53  41  01FCF                                                                         ;
                                                            01FD2         .BLKB    2
                                                  00000012  01FD4 P.AEU:  .LONG    18                                                     ;
                                                  00000000' 01FD8         .ADDRESS P.AEV                                                  ;
21  28  3D  4D  52  4F  46  5F  50  55  54  45  53  2F  20  01FDC P.AEX:  .ASCII   \ /SETUP_FORM=(!AS)\                                   ;
                                                29  53  41  01FEB                                                                         ;
                                                            01FEE         .BLKB    2
                                                  00000012  01FF0 P.AEW:  .LONG    18                                                     ;
                                                  00000000' 01FF4         .ADDRESS P.AEX                                                  ;
            44  45  45  46  5F  54  45  45  48  53  2F  20  01FF8 P.AEZ:  .ASCII   \ /SHEET_FEED\                                         ;
                                                  0000000C  02004 P.AEY:  .LONG    12                                                     ;
                                                  00000000' 02008         .ADDRESS P.AEZ                                                  ;
                                45  43  41  50  53  2F  20  0200C P.AFB:  .ASCII   \ /SPACE\                                              ;
                                                            02013         .BLKB    1
                                                  00000007  02014 P.AFA:  .LONG    7                                                      ;
                                                  00000000' 02018         .ADDRESS P.AFB                                                  ;
                        52  45  4C  49  41  52  54  2F  20  0201C P.AFD:  .ASCII   \ /TRAILER\                                            ;
                                                            02025         .BLKB    3
                                                  00000009  02028 P.AFC:  .LONG    9                                                      ;
                                                  00000000' 0202C         .ADDRESS P.AFD                                                  ;
                    45  54  41  43  4E  55  52  54  2F  20  02030 P.AFF:  .ASCII   \ /TRUNCATE\                                           ;
                                                            0203A         .BLKB    2
                                                  0000000A  0203C P.AFE:  .LONG    10                                                     ;
                                                  00000000' 02040         .ADDRESS P.AFF                                                  ;
                4C  55  21  3D  48  54  44  49  57  2F  20  02044 P.AFH:  .ASCII   \ /WIDTH=!UL\                                          ;
                                                            0204F         .BLKB    1
                                                  0000000B  02050 P.AFG:  .LONG    11                                                     ;
                                                  00000000' 02054         .ADDRESS P.AFH                                                  ;
                                    50  41  52  57  2F  20  02058 P.AFJ:  .ASCII   \ /WRAP\                                               ;
                                                            0205E         .BLKB    2
                                                  00000006  02060 P.AFI:  .LONG    6                                                      ;
                                                  00000000' 02064         .ADDRESS P.AFJ                                                  ;
                                                    2C  01  02068 P.AFK:  .ASCII   <1>\,\                                                 ;
                                                    2C  01  0206A P.AFL:  .ASCII   <1>\,\                                                 ;
                                                    2C  01  0206C P.AFM:  .ASCII   <1>\,\                                                 ;
                                                    29  01  0206E P.AFN:  .ASCII   <1>\)\                                                 ;
                                                    DATE_FORMAT=           P.ADA
                                                    AFTER_TIME_FORMAT=     P.ADC
                                                    BURST_FORMAT=          P.ADE
                                                    CHARACTERISTICS_FORMAT=
                                                                           P.ADG
                                                    FILE_COPIES_FORMAT=    P.ADI
                                                    FEED_FORMAT=           P.ADK
                                                    FLAG_FORMAT=           P.ADM
                                                    FORM_FORMAT=           P.ADO
                                                    HEADER_FORMAT=         P.ADQ
                                                    JOB_COUNT_FORMAT=      P.ADS
                                                    LENGTH_FORMAT=         P.ADU
                                                    LIBRARY_FORMAT=        P.ADW
                                                    MARGIN_FORMAT=         P.ADY
                                                    TOP_FORMAT=            P.AEA
                                                    BOTTOM_FORMAT=         P.AEC
                                                    LEFT_FORMAT=           P.AEE
                                                    RIGHT_FORMAT=          P.AEG
                                                    NOFEED_FORMAT=         P.AEI
                                                    SETUP_PAGE_FORMAT=     P.AEK
                                                    PAGES_FORMAT=          P.AEM
                                                    PARAMETER_FORMAT=      P.AEO
                                                    PASSALL_FORMAT=        P.AEQ
                                                    PUNCTUATION_FORMAT=    P.AES
                                                    SETUP_FILE_FORMAT=     P.AEU
                                                    SETUP_FORM_FORMAT=     P.AEW
                                                    SHEET_FORMAT=          P.AEY
                                                    SPACE_FORMAT=          P.AFA
                                                    TRAILER_FORMAT=        P.AFC
                                                    TRUNCATE_FORMAT=       P.AFE
                                                    WIDTH_FORMAT=          P.AFG
                                                    WRAP_FORMAT=           P.AFI


                                     07FC 00000 GET_QUALIFIERS:
                                                           .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10                ; 4690
                    5A       FF42  CF   9E 00002           MOVAB    PUNCTUATION_FORMAT, R10                         ;
                    59 00000000G   00   9E 00007           MOVAB    SYS$FAO, R9                                     ;
                    5E         AC  AE   9E 0000E           MOVAB    -84(SP), SP                                     ;
                                   56   D4 00012           CLRL     PUNC_FLAG                                       ; 4695
                               08  AE   D4 00014           CLRL     CURRENT_LEN                                     ;
                 14    AE    0200  8F   3C 00017           MOVZWL   #512, STRING_PTR                                ; 4810
                       50      08  AC   D0 0001D           MOVL     STR_DESC, R0                                    ; 4811
                 18    AE      04  A0   D0 00021           MOVL     4(R0), STRING_PTR+4                             ;
                       55      0C  AC   D0 00026           MOVL     RET_LEN, R5                                     ; 4813
                                   65   B4 0002A           CLRW     (R5)                                            ;
                 4C    AE          11   D0 0002C           MOVL     #17, AFT_DATE_PTR                               ; 4827
                 50    AE      30  AE   9E 00030           MOVAB    AFT_BUFF, AFT_DATE_PTR+4                        ; 4828
                       54      04  AC   D0 00035           MOVL     SCB, R4                                         ; 4833
                               24  A4   9F 00039           PUSHAB   36(R4)                                          ;
                               50  AE   9F 0003C           PUSHAB   AFT_DATE_PTR                                    ;
                               0C  AE   9F 0003F           PUSHAB   TEMP_LEN                                        ;
                             FDF0  CA   9F 00042           PUSHAB   DATE_FORMAT                                     ;
                       69          04   FB 00046           CALLS    #4, SYS$FAO                                     ;
                 44    AE          11   D0 00049           MOVL     #17, TEMP_PTR                                   ; 4835
                 48    AE      1C  AE   9E 0004D           MOVAB    TEMP_BUFF, TEMP_PTR+4                           ; 4836
                             015C  C4   9F 00052           PUSHAB   348(R4)                                         ; 4841
```

```
                           48      AE  9F 00056          PUSHAB  TEMP_PTR                                  ;
                           0C      AE  9F 00059          PUSHAB  TEMP_LEN                                  ;
                         FDF0      CA  9F 0005C          PUSHAB  DATE_FORMAT                               ;
                   69      04          FB 00060          CALLS   #4, SYS$FAO                               ;
   50    BE        48  BE      04  AE  29 00063          CMPC3   TEMP_LEN, @TEMP_PTR+4, @AFT_DATE_PTR+4    ; 4843
                           27          13 0006A          BEQL    1$                                        ;
                           24      A4  9F 0006C          PUSHAB  36(R4)                                    ; 4851
                           18      AE  9F 0006F          PUSHAB  STRING_PTR                                ;
                           10      AE  9F 00072          PUSHAB  CURRENT_LEN                               ;
                         FE08      CA  9F 00075          PUSHAB  AFTER_TIME_FORMAT                         ;
                   69      04          FB 00079          CALLS   #4, SYS$FAO                               ;
                   65      08      AE  A0 0007C          ADDW2   CURRENT_LEN, (R5)                         ; 4853
            18     AE      08      AE  C0 00080          ADDL2   CURRENT_LEN, STRING_PTR+4                 ; 4854
            14     AE              65  3C 00085          MOVZWL  (R5), STRING_PTR                          ; 4855
   14    AE 00000200  8F   14      AE  C3 00089          SUBL3   STRING_PTR, #512, STRING_PTR              ;
                   58    0154      C4  9E 00093 1$:      MOVAB   340(R4), R8                               ; 4858
                   24      68          E9 00098          BLBC    (R8), 2$                                  ;
                           14      AE  9F 0009B          PUSHAB  STRING_PTR                                ; 4864
                           0C      AE  9F 0009E          PUSHAB  CURRENT_LEN                               ;
                         FE18      CA  9F 000A1          PUSHAB  BURST_FORMAT                              ;
                   69      03          FB 000A5          CALLS   #3, SYS$FAO                               ;
                   65      08      AE  A0 000A8          ADDW2   CURRENT_LEN, (R5)                         ; 4866
            18     AE      08      AE  C0 000AC          ADDL2   CURRENT_LEN, STRING_PTR+4                 ; 4867
            14     AE              65  3C 000B1          MOVZWL  (R5), STRING_PTR                          ; 4868
   14    AE 00000200  8F   14      AE  C3 000B5          SUBL3   STRING_PTR, #512, STRING_PTR              ;
                           34      A4  B5 000BF 2$:      TSTW    52(R4)                                    ; 4871
                           27          12 000C2          BNEQ    3$                                        ;
                           34      A4  9F 000C4          PUSHAB  52(R4)                                    ; 4878
                           18      AE  9F 000C7          PUSHAB  STRING_PTR                                ;
                           10      AE  9F 000CA          PUSHAB  CURRENT_LEN                               ;
                         FE38      CA  9F 000CD          PUSHAB  CHARACTERISTICS_FORMAT                    ;
                   69      04          FB 000D1          CALLS   #4, SYS$FAO                               ;
                   65      08      AE  A0 000D4          ADDW2   CURRENT_LEN, (R5)                         ; 4880
            18     AE      08      AE  C0 000D8          ADDL2   CURRENT_LEN, STRING_PTR+4                 ; 4881
            14     AE              65  3C 000DD          MOVZWL  (R5), STRING_PTR                          ; 4882
   14    AE 00000200  8F   14      AE  C3 000E1          SUBL3   STRING_PTR, #512, STRING_PTR              ;
                   57    0124      C4  9E 000EB 3$:      MOVAB   292(R4), R7                               ; 4887
         24        67      02          E1 000F0          BBC     #2, (R7), 4$                              ;
                           14      AE  9F 000F4          PUSHAB  STRING_PTR                                ; 4893
                           0C      AE  9F 000F7          PUSHAB  CURRENT_LEN                               ;
                         FE68      CA  9F 000FA          PUSHAB  FEED_FORMAT                               ;
                   69      03          FB 000FE          CALLS   #3, SYS$FAO                               ;
                   65      08      AE  A0 00101          ADDW2   CURRENT_LEN, (R5)                         ; 4895
            18     AE      08      AE  C0 00105          ADDL2   CURRENT_LEN, STRING_PTR+4                 ; 4896
            14     AE              65  3C 0010A          MOVZWL  (R5), STRING_PTR                          ; 4897
   14    AE 00000200  8F   14      AE  C3 0010E          SUBL3   STRING_PTR, #512, STRING_PTR              ;
                   01      64      A4  D1 00118 4$:      CMPL    100(R4), #1                               ; 4900
                           2B          15 0011C          BLEQ    5$                                        ;
                           64      A4  DD 0011E          PUSHL   100(R4)                                   ; 4909
                   7E      64      A4  7D 00121          MOVQ    100(R4), -(SP)                            ;
                           20      AE  9F 00125          PUSHAB  STRING_PTR                                ;
                           18      AE  9F 00128          PUSHAB  CURRENT_LEN                               ;
                         FE58      CA  9F 0012B          PUSHAB  FILE_COPIES_FORMAT                        ;
                   69      06          FB 0012F          CALLS   #6, SYS$FAO                               ;
                   65      08      AE  A0 00132          ADDW2   CURRENT_LEN, (R5)                         ; 4911
            18     AE      08      AE  C0 00136          ADDL2   CURRENT_LEN, STRING_PTR+4                 ; 4912
            14     AE              65  3C 0013B          MOVZWL  (R5), STRING_PTR                          ; 4913
```

```
14  AE 00000200   8F       14   AE  C3 0013F        SUBL3    STRING_PTR, #512, STRING_PTR        ;
    24            68            01  E1 00149 5$:    BBC      #1, (R8), 6$                        ; 4916
                           14   AE  9F 0014D        PUSHAB   STRING_PTR                          ; 4922
                           0C   AE  9F 00150        PUSHAB   CURRENT_LEN                         ;
                         FE78   CA  9F 00153        PUSHAB   FLAG_FORMAT                         ;
                  69            03  FB 00157        CALLS    #3, SYS$FAO                         ;
                  65       08   AE  A0 0015A        ADDW2    CURRENT_LEN, (R5)                   ; 4924
             18   AE       08   AE  C0 0015E        ADDL2    CURRENT_LEN, STRING_PTR+4           ; 4925
             14   AE            65  3C 00163        MOVZWL   (R5), STRING_PTR                    ; 4926
14  AE 00000200   8F       14   AE  C3 00167        SUBL3    STRING_PTR, #512, STRING_PTR        ;
                         0084   C4  9F 00171 6$:    PUSHAB   132(R4)                             ; 4932
                           18   AE  9F 00175        PUSHAB   STRING_PTR                          ;
                           10   AE  9F 00178        PUSHAB   CURRENT_LEN                         ;
                         FE8C   CA  9F 0017B        PUSHAB   FORM_FORMAT                         ;
                  69            04  FB 0017F        CALLS    #4, SYS$FAO                         ;
                  65       08   AE  A0 00182        ADDW2    CURRENT_LEN, (R5)                   ; 4934
             18   AE       08   AE  C0 00186        ADDL2    CURRENT_LEN, STRING_PTR+4           ; 4935
             14   AE            65  3C 0018B        MOVZWL   (R5), STRING_PTR                    ; 4936
14  AE 00000200   8F       14   AE  C3 0018F        SUBL3    STRING_PTR, #512, STRING_PTR        ;
           0200   8F            65  B1 00199        CMPW     (R5), #512                          ; 4938
                                2D  1A 0019E        BGTRU    8$                                  ;
    24            67            01  E1 001A0        BBC      #1, (R7), 7$                        ; 4945
                           14   AE  9F 001A4        PUSHAB   STRING_PTR                          ; 4951
                           0C   AE  9F 001A7        PUSHAB   CURRENT_LEN                         ;
                         FE9C   CA  9F 001AA        PUSHAB   HEADER_FORMAT                       ;
                  69            03  FB 001AE        CALLS    #3, SYS$FAO                         ;
                  65       08   AE  A0 001B1        ADDW2    CURRENT_LEN, (R5)                   ; 4953
             18   AE       08   AE  C0 001B5        ADDL2    CURRENT_LEN, STRING_PTR+4           ; 4954
             14   AE            65  3C 001BA        MOVZWL   (R5), STRING_PTR                    ; 4955
14  AE 00000200   8F       14   AE  C3 001BE        SUBL3    STRING_PTR, #512, STRING_PTR        ;
           0200   8F            65  B1 001C8 7$:    CMPW     (R5), #512                          ; 4958
                                03  1B 001CD 8$:    BLEQU    9$                                  ;
                              051C  31 001CF        BRW      45$                                 ;
                  50     00A0   C4  D0 001D2 9$:    MOVL     160(R4), R0                         ; 4965
                  01            50  D1 001D7        CMPL     R0, #1                              ;
                                2C  15 001DA        BLEQ     10$                                 ;
                                50  DD 001DC        PUSHL    R0                                  ; 4974
                         00A4   C4  DD 001DE        PUSHL    164(R4)                             ;
                                50  DD 001E2        PUSHL    R0                                  ;
                           20   AE  9F 001E4        PUSHAB   STRING_PTR                          ;
                           18   AE  9F 001E7        PUSHAB   CURRENT_LEN                         ;
                         FEC0   CA  9F 001EA        PUSHAB   JOB_COUNT_FORMAT                    ;
                  69            06  FB 001EE        CALLS    #6, SYS$FAO                         ;
                  65       08   AE  A0 001F1        ADDW2    CURRENT_LEN, (R5)                   ; 4976
             18   AE       08   AE  C0 001F5        ADDL2    CURRENT_LEN, STRING_PTR+4           ; 4977
             14   AE            65  3C 001FA        MOVZWL   (R5), STRING_PTR                    ; 4978
14  AE 00000200   8F       14   AE  C3 001FE        SUBL3    STRING_PTR, #512, STRING_PTR        ;
             0C   AE 020E0000   8F  D0 00208 10$:   MOVL     #34471936, X_DESC                   ; 4987
                           10   AE  D4 00210        CLRL     X_DESC+4                            ;
                           0C   AE  9F 00213        PUSHAB   LEN                                 ; 4989
             04   AE            11  D0 00216        MOVL     #17, 4(SP)                          ; 4988
                           04   AE  9F 0021A        PUSHAB   4(SP)                               ;
                                54  DD 0021D        PUSHL    R4                                  ;
      00000000G   00            03  FB 0021F        CALLS    #3, PSM$READ_ITEM_DX                ;
                           10   BE  DD 00226        PUSHL    @LEN+4                              ; 4995
                           18   AE  9F 00229        PUSHAB   STRING_PTR                          ;
                           10   AE  9F 0022C        PUSHAB   CURRENT_LEN                         ;
```

```
                             FED4   CA  9F 0022F         PUSHAB  LENGTH_FORMAT                           ;
                       69           04  FB 00233         CALLS   #4, SYS$FAO                             ;
                       65      08   AE  A0 00236         ADDW2   CURRENT_LEN, (R5)                       ; 4997
                 18    AE      08   AE  C0 0023A         ADDL2   CURRENT_LEN, STRING_PTR+4               ; 4998
                 14    AE           65  3C 0023F         MOVZWL  (R5), STRING_PTR                        ; 4999
   14   AE 00000200    8F      14   AE  C3 00243         SUBL3   STRING_PTR, #512, STRING_PTR            ;
                             00C0   C4  9F 0024D         PUSHAB  192(R4)                                 ; 5008
                               18   AE  9F 00251         PUSHAB  STRING_PTR                              ;
                               10   AE  9F 00254         PUSHAB  CURRENT_LEN                             ;
                             FEEC   CA  9F 00257         PUSHAB  LIBRARY_FORMAT                          ;
                       69           04  FB 0025B         CALLS   #4, SYS$FAO                             ;
                       65      08   AE  A0 0025E         ADDW2   CURRENT_LEN, (R5)                       ; 5010
                 18    AE      08   AE  C0 00262         ADDL2   CURRENT_LEN, STRING_PTR+4               ; 5011
                 14    AE           65  3C 00267         MOVZWL  (R5), STRING_PTR                        ; 5012
   14   AE 00000200    8F      14   AE  C3 0026B         SUBL3   STRING_PTR, #512, STRING_PTR            ;
               0200    8F           65  B1 00275         CMPW    (R5), #512                              ; 5014
                                    03  1B 0027A         BLEQU   11$                                     ;
                                  046F  31 0027C         BRW     45$                                     ;
                       52    0164   C4  D0 0027F 11$:    MOVL    356(R4), R2                             ; 5021
                                    53  D4 00284         CLRL    R3                                      ;
                                    52  D5 00286         TSTL    R2                                      ;
                                    04  13 00288         BEQL    12$                                     ;
                                    53  D6 0028A         INCL    R3                                      ;
                                    14  11 0028C         BRB     13$                                     ;
                               30   A4  D5 0028E 12$:    TSTL    48(R4)                                  ; 5022
                                    0F  12 00291         BNEQ    13$                                     ;
                             00BC   C4  D5 00293         TSTL    188(R4)                                 ; 5023
                                    09  12 00297         BNEQ    13$                                     ;
                             0148   C4  D5 00299         TSTL    328(R4)                                 ; 5024
                                    03  12 0029D         BNEQ    13$                                     ;
                                  0196  31 0029F         BRW     21$                                     ;
                               14   AE  9F 002A2 13$:    PUSHAB  STRING_PTR                              ; 5031
                               0C   AE  9F 002A5         PUSHAB  CURRENT_LEN                             ;
                             FF00   CA  9F 002A8         PUSHAB  MARGIN_FORMAT                           ;
                       69           03  FB 002AC         CALLS   #3, SYS$FAO                             ;
                       65      08   AE  A0 002AF         ADDW2   CURRENT_LEN, (R5)                       ; 5033
                 18    AE      08   AE  C0 002B3         ADDL2   CURRENT_LEN, STRING_PTR+4               ; 5034
                 14    AE           65  3C 002B8         MOVZWL  (R5), STRING_PTR                        ; 5035
   14   AE 00000200    8F      14   AE  C3 002BC         SUBL3   STRING_PTR, #512, STRING_PTR            ;
                       29           53  E9 002C6         BLBC    R3, 14$                                 ; 5037
                                    52  DD 002C9         PUSHL   R2                                      ; 5044
                               18   AE  9F 002CB         PUSHAB  STRING_PTR                              ;
                               10   AE  9F 002CE         PUSHAB  CURRENT_LEN                             ;
                             FF10   CA  9F 002D1         PUSHAB  TOP_FORMAT                              ;
                       69           04  FB 002D5         CALLS   #4, SYS$FAO                             ;
                       56           01  D0 002D8         MOVL    #1, PUNC_FLAG                           ; 5046
                       65      08   AE  A0 002DB         ADDW2   CURRENT_LEN, (R5)                       ; 5047
                 18    AE      08   AE  C0 002DF         ADDL2   CURRENT_LEN, STRING_PTR+4               ; 5048
                 14    AE           65  3C 002E4         MOVZWL  (R5), STRING_PTR                        ; 5049
   14   AE 00000200    8F      14   AE  C3 002E8         SUBL3   STRING_PTR, #512, STRING_PTR            ;
                                    52  D4 002F2 14$:    CLRL    R2                                      ; 5052
                               30   A4  D5 002F4         TSTL    48(R4)                                  ;
                                    2B  13 002F7         BEQL    15$                                     ;
                                    52  D6 002F9         INCL    R2                                      ;
                       26           56  E9 002FB         BLBC    PUNC_FLAG, 15$                          ; 5053
                             00B0   CA  9F 002FE         PUSHAB  P.AFR                                   ; 5061
                               18   AE  9F 00302         PUSHAB  STRING_PTR                              ;
```

```
                                    10    AE   9F 00305          PUSHAB  CURRENT_LEN                          ;
                                          5A   DD 00308          PUSHL   R10                                  ;
                          69              04   FB 0030A          CALLS   #4, SYS$FAO                          ;
                          65        08    AE   A0 0030D          ADDW2   CURRENT_LEN, (R5)                    ; 5063
                   18     AE        08    AE   C0 00311          ADDL2   CURRENT_LEN, STRING_PTR+4            ; 5064
                   14     AE              65   3C 00316          MOVZWL  (R5), STRING_PTR                     ; 5065
   14   AE 00000200       8F        14    AE   C3 0031A          SUBL3   STRING_PTR, #512, STRING_PTR
                          2A              52   E9 00324  15$:    BLBC    R2, 16$                              ; 5068
                                    30    A4   DD 00327          PUSHL   48(R4)                               ; 5076
                                    18    AE   9F 0032A          PUSHAB  STRING_PTR                           ;
                                    10    AE   9F 0032D          PUSHAB  CURRENT_LEN                          ;
                                  FF24    CA   9F 00330          PUSHAB  BOTTOM_FORMAT                        ;
                          69              04   FB 00334          CALLS   #4, SYS$FAO                          ;
                          56              01   D0 00337          MOVL    #1, PUNC_FLAG                        ; 5078
                          65        08    AE   A0 0033A          ADDW2   CURRENT_LEN, (R5)                    ; 5079
                   18     AE        08    AE   C0 0033E          ADDL2   CURRENT_LEN, STRING_PTR+4            ; 5080
                   14     AE              65   3C 00343          MOVZWL  (R5), STRING_PTR                     ; 5081
   14   AE 00000200       8F        14    AE   C3 00347          SUBL3   STRING_PTR, #512, STRING_PTR
                          52      00BC    C4   D0 00351  16$:    MOVL    188(R4), R2                          ; 5084
                                          53   D4 00356          CLRL    R3                                   ;
                                          52   D5 00358          TSTL    R2                                   ;
                                          2B   13 0035A          BEQL    17$                                  ;
                                          53   D6 0035C          INCL    R3                                   ;
                          26              56   E9 0035E          BLBC    PUNC_FLAG, 17$                       ; 5085
                                  00B2    CA   9F 00361          PUSHAB  P.AF[                                ; 5093
                                    18    AE   9F 00365          PUSHAB  STRING_PTR                           ;
                                    10    AE   9F 00368          PUSHAB  CURRENT_LEN                          ;
                                          5A   DD 0036B          PUSHL   R10                                  ;
                          69              04   FB 0036D          CALLS   #4, SYS$FAO                          ;
                          65        08    AE   A0 00370          ADDW2   CURRENT_LEN, (R5)                    ; 5094
                   18     AE        08    AE   C0 00374          ADDL2   CURRENT_LEN, STRING_PTR+4            ; 5095
                   14     AE              65   3C 00379          MOVZWL  (R5), STRING_PTR                     ; 5096
   14   AE 00000200       8F        14    AE   C3 0037D          SUBL3   STRING_PTR, #512, STRING_PTR
                          29              53   E9 00387  17$:    BLBC    R3, 18$                              ; 5099
                                          52   DD 0038A          PUSHL   R2                                   ; 5106
                                    18    AE   9F 0038C          PUSHAB  STRING_PTR                           ;
                                    10    AE   9F 0038F          PUSHAB  CURRENT_LEN                          ;
                                  FF34    CA   9F 00392          PUSHAB  LEFT_FORMAT                          ;
                          69              04   FB 00396          CALLS   #4, SYS$FAO                          ;
                          56              01   D0 00399          MOVL    #1, PUNC_FLAG                        ; 5108
                          65        08    AE   A0 0039C          ADDW2   CURRENT_LEN, (R5)                    ; 5109
                   18     AE        08    AE   C0 003A0          ADDL2   CURRENT_LEN, STRING_PTR+4            ; 5110
                   14     AE              65   3C 003A5          MOVZWL  (R5), STRING_PTR                     ; 5111
   14   AE 00000200       8F        14    AE   C3 003A9          SUBL3   STRING_PTR, #512, STRING_PTR
                          52      0148    C4   D0 003B3  18$:    MOVL    328(R4), R2                          ; 5114
                                          53   D4 003B8          CLRL    R3                                   ;
                                          52   D5 003BA          TSTL    R2                                   ;
                                          2B   13 003BC          BEQL    19$                                  ;
                                          53   D6 003BE          INCL    R3                                   ;
                          26              56   E9 003C0          BLBC    PUNC_FLAG, 19$                       ; 5115
                                  00B4    CA   9F 003C3          PUSHAB  P.AFM                                ; 5123
                                    18    AE   9F 003C7          PUSHAB  STRING_PTR                           ;
                                    10    AE   9F 003CA          PUSHAB  CURRENT_LEN                          ;
                                          5A   DD 003CD          PUSHL   R10                                  ;
                          69              04   FB 003CF          CALLS   #4, SYS$FAO                          ;
                          65        08    AE   A0 003D2          ADDW2   CURRENT_LEN, (R5)                    ; 5125
                   18     AE        08    AE   C0 003D6          ADDL2   CURRENT_LEN, STRING_PTR+4            ; 5126
                   14   AE                  65   3C 003DB          MOVZWL   (R5), STRING_PTR                       ; 5127
    14   AE 00000200   8F          14   AE   C3 003DF          SUBL3    STRING_PTR, #512, STRING_PTR           ;
                       26                   53   E9 003E9 19$:     BLBC     R3, 20$                                ; 5130
                                            52   DD 003EC          PUSHL    R2                                     ; 5137
                                   18   AE   9F 003EE          PUSHAB   STRING_PTR                             ;
                                   10   AE   9F 003F1          PUSHAB   CURRENT_LEN                            ;
                                 FF48   CA   9F 003F4          PUSHAB   RIGHT_FORMAT                           ;
                       69                   04   FB 003F8          CALLS    #4, SYS$FAO                            ;
                       65          08   AE   A0 003FB          ADDW2    CURRENT_LEN, (R5)                      ; 5139
                   18   AE          08   AE   C0 003FF          ADDL2    CURRENT_LEN, STRING_PTR+4              ; 5140
                   14   AE                  65   3C 00404          MOVZWL   (R5), STRING_PTR                       ; 5141
    14   AE 00000200   8F          14   AE   C3 00408          SUBL3    STRING_PTR, #512, STRING_PTR           ;
                                 00B6   CA   9F 00412 20$:     PUSHAB   P.AFN                                  ; 5149
                                   18   AE   9F 00416          PUSHAB   STRING_PTR                             ;
                                   10   AE   9F 00419          PUSHAB   CURRENT_LEN                            ;
                                            5A   DD 0041C          PUSHL    R10                                    ;
                       69                   04   FB 0041E          CALLS    #4, SYS$FAO                            ;
                       65          08   AE   A0 00421          ADDW2    CURRENT_LEN, (R5)                      ; 5151
                   18   AE          08   AE   C0 00425          ADDL2    CURRENT_LEN, STRING_PTR+4              ; 5152
                   14   AE                  65   3C 0042A          MOVZWL   (R5), STRING_PTR                       ; 5153
    14   AE 00000200   8F          14   AE   C3 0042E          SUBL3    STRING_PTR, #512, STRING_PTR           ;
                0200   8F                   65   B1 00438 21$:     CMPW     (R5), #512                             ; 5156
                                            69   1A 0043D          BGTRU    25$                                    ;
          24           67                   02   E0 0043F          BBS      #2, (R7), 22$                          ; 5163
                                   14   AE   9F 00443          PUSHAB   STRING_PTR                             ; 5169
                                   0C   AE   9F 00446          PUSHAB   CURRENT_LEN                            ;
                                 FF58   CA   9F 00449          PUSHAB   NOFEED_FORMAT                          ;
                       69                   03   FB 0044D          CALLS    #3, SYS$FAO                            ;
                       65          08   AE   A0 00450          ADDW2    CURRENT_LEN, (R5)                      ; 5171
                   18   AE          08   AE   C0 00454          ADDL2    CURRENT_LEN, STRING_PTR+4              ; 5172
                   14   AE                  65   3C 00459          MOVZWL   (R5), STRING_PTR                       ; 5173
    14   AE 00000200   8F          14   AE   C3 0045D          SUBL3    STRING_PTR, #512, STRING_PTR           ;
                0200   8F                   65   B1 00467 22$:     CMPW     (R5), #512                             ; 5176
                                            3A   1A 0046C          BGTRU    25$                                    ;
                                   74   A4   D5 0046E          TSTL     116(R4)                                ; 5183
                                            06   12 00471          BNEQ     23$                                    ;
                                 00B8   C4   D5 00473          TSTL     184(R4)                                ; 5184
                                            2A   13 00477          BEQL     24$                                    ;
                                 00B8   C4   DD 00479 23$:     PUSHL    184(R4)                                ; 5192
                                   74   A4   DD 0047D          PUSHL    116(R4)                                ;
                                   1C   AE   9F 00480          PUSHAB   STRING_PTR                             ;
                                   14   AE   9F 00483          PUSHAB   CURRENT_LEN                            ;
                                   90   AA   9F 00486          PUSHAB   PAGES_FORMAT                           ;
                       69                   05   FB 00489          CALLS    #5, SYS$FAO                            ;
                       65          08   AE   A0 0048C          ADDW2    CURRENT_LEN, (R5)                      ; 5194
                   18   AE          08   AE   C0 00490          ADDL2    CURRENT_LEN, STRING_PTR+4              ; 5195
                   14   AE                  65   3C 00495          MOVZWL   (R5), STRING_PTR                       ; 5196
    14   AE 00000200   8F          14   AE   C3 00499          SUBL3    STRING_PTR, #512, STRING_PTR           ;
                0200   8F                   65   B1 004A3 24$:     CMPW     (R5), #512                             ; 5199
                                            78   1A 004A8 25$:     BGTRU    28$                                    ;
                                 00E4   C4   B5 004AA          TSTW     228(R4)                                ; 5206
                                            2A   12 004AE          BNEQ     26$                                    ;
                                 00EC   C4   B5 004B0          TSTW     236(R4)                                ; 5207
                                            24   12 004B4          BNEQ     26$                                    ;
                                 00F4   C4   B5 004B6          TSTW     244(R4)                                ; 5208
                                            1E   12 004BA          BNEQ     26$                                    ;
                                 00FC   C4   B5 004BC          TSTW     252(R4)                                ; 5209
```

```
                                          18  12 004C0          BNEQ    26$                                         ;
                                    0104  C4  B5 004C2          TSTW    260(R4)                                     ; 5210
                                          12  12 004C6          BNEQ    26$                                         ;
                                    010C  C4  B5 004C8          TSTW    268(R4)                                     ; 5211
                                          0C  12 004CC          BNEQ    26$                                         ;
                                    0114  C4  B5 004CE          TSTW    276(R4)                                     ; 5212
                                          06  12 004D2          BNEQ    26$                                         ;
                                    011C  C4  B5 004D4          TSTW    284(R4)                                     ; 5213
                                          43  13 004D8          BEQL    27$                                         ;
                                    011C  C4  9F 004DA 26$:     PUSHAB  284(R4)                                     ; 5228
                                    0114  C4  9F 004DE          PUSHAB  276(R4)                                     ;
                                    010C  C4  9F 004E2          PUSHAB  268(R4)                                     ;
                                    0104  C4  9F 004E6          PUSHAB  260(R4)                                     ;
                                    00FC  C4  9F 004EA          PUSHAB  252(R4)                                     ;
                                    00F4  C4  9F 004EE          PUSHAB  244(R4)                                     ;
                                    00EC  C4  9F 004F2          PUSHAB  236(R4)                                     ;
                                    00E4  C4  9F 004F6          PUSHAB  228(R4)                                     ;
                                      34  AE  9F 004FA          PUSHAB  STRING_PTR                                  ;
                                      2C  AE  9F 004FD          PUSHAB  CURRENT_LEN                                 ;
                                      E0  AA  9F 00500          PUSHAB  PARAMETER_FORMAT                            ;
                           69             0B  FB 00503          CALLS   #11, SYS$FAO                                ;
                           65         08  AE  A0 00506          ADDW2   CURRENT_LEN, (R5)                           ; 5230
                   18      AE         08  AE  C0 0050A          ADDL2   CURRENT_LEN, STRING_PTR+4                   ; 5231
                   14      AE             65  3C 0050F          MOVZWL  (R5), STRING_PTR                            ; 5232
     14    AE 00000200     8F         14  AE  C3 00513          SUBL3   STRING_PTR, #512, STRING_PTR                ;
                 0200      8F             65  B1 0051D 27$:     CMPW    (R5), #512                                  ; 5235
                                          5E  1A 00522 28$:     BGTRU   31$                                         ;
     23                    67             03  E1 00524          BBC     #3, (R7), 29$                               ; 5242
                                      14  AE  9F 00528          PUSHAB  STRING_PTR                                  ; 5248
                                      0C  AE  9F 0052B          PUSHAB  CURRENT_LEN                                 ;
                                      F4  AA  9F 0052E          PUSHAB  PASSALL_FORMAT                              ;
                           69             03  FB 00531          CALLS   #3, SYS$FAO                                 ;
                           65         08  AE  A0 00534          ADDW2   CURRENT_LEN, (R5)                           ; 5250
                   18      AE         08  AE  C0 00538          ADDL2   CURRENT_LEN, STRING_PTR+4                   ; 5251
                   14      AE             65  3C 0053D          MOVZWL  (R5), STRING_PTR                            ; 5252
     14    AE 00000200     8F         14  AE  C3 00541          SUBL3   STRING_PTR, #512, STRING_PTR                ;
                 0200      8F             65  B1 0054B 29$:     CMPW    (R5), #512                                  ; 5255
                                          30  1A 00550          BGTRU   31$                                         ;
                                      6C  A4  B5 00552          TSTW    108(R4)                                     ; 5262
                                          26  13 00555          BEQL    30$                                         ;
                                      6C  A4  9F 00557          PUSHAB  108(R4)                                     ; 5269
                                      18  AE  9F 0055A          PUSHAB  STRING_PTR                                  ;
                                      10  AE  9F 0055D          PUSHAB  CURRENT_LEN                                 ;
                                      1C  AA  9F 00560          PUSHAB  SETUP_FILE_FORMAT                           ;
                           69             04  FB 00563          CALLS   #4, SYS$FAO                                 ;
                           65         08  AE  A0 00566          ADDW2   CURRENT_LEN, (R5)                           ; 5271
                   18      AE         08  AE  C0 0056A          ADDL2   CURRENT_LEN, STRING_PTR+4                   ; 5272
                   14      AE             65  3C 0056F          MOVZWL  (R5), STRING_PTR                            ; 5273
     14    AE 00000200     8F         14  AE  C3 00573          SUBL3   STRING_PTR, #512, STRING_PTR                ;
                 0200      8F             65  B1 0057D 30$:     CMPW    (R5), #512                                  ; 5276
                                          5E  1A 00582 31$:     BGTRU   34$                                         ;
                                      7C  A4  B5 00584          TSTW    124(R4)                                     ; 5283
                                          26  13 00587          BEQL    32$                                         ;
                                      7C  A4  9F 00589          PUSHAB  124(R4)                                     ; 5290
                                      18  AE  9F 0058C          PUSHAB  STRING_PTR                                  ;
                                      10  AE  9F 0058F          PUSHAB  CURRENT_LEN                                 ;
                                      38  AA  9F 00592          PUSHAB  SETUP_FORM_FORMAT                           ;
```

```
                       69              04  FB 00595        CALLS   #4, SYS$FAO                                ;
                       65        08    AE  A0 00598        ADDW2   CURRENT_LEN, (R5)                          ; 5292
              18       AE        08    AE  C0 0059C        ADDL2   CURRENT_LEN, STRING_PTR+4                  ; 5293
              14       AE              65  3C 005A1        MOVZWL  (R5), STRING_PTR                           ; 5294
  14  AE 00000200      8F        14    AE  C3 005A5        SUBL3   STRING_PTR, #512, STRING_PTR               ;
                               00DC    C4  B5 005AF 32$:   TSTW    220(R4)                                    ; 5297
                                       28  13 005B3        BEQL    33$                                        ;
                               00DC    C4  9F 005B5        PUSHAB  220(R4)                                    ; 5304
                                 18    AE  9F 005B9        PUSHAB  STRING_PTR                                 ;
                                 10    AE  9F 005BC        PUSHAB  CURRENT_LEN                                ;
                               FF74    CA  9F 005BF        PUSHAB  SETUP_PAGE_FORMAT                          ;
                       69              04  FB 005C3        CALLS   #4, SYS$FAO                                ;
                       65        08    AE  A0 005C6        ADDW2   CURRENT_LEN, (R5)                          ; 5306
              18       AE        08    AE  C0 005CA        ADDL2   CURRENT_LEN, STRING_PTR+4                  ; 5307
              14       AE              65  3C 005CF        MOVZWL  (R5), STRING_PTR                           ; 5308
  14  AE 00000200      8F        14    AE  C3 005D3        SUBL3   STRING_PTR, #512, STRING_PTR               ;
            0200       8F              65  B1 005DD 33$:   CMPW    (R5), #512                                 ; 5311
                                       59  1A 005E2 34$:   BGTRU   37$                                        ;
  23                   67              05  E1 005E4        BBC     #5, (R7), 35$                              ; 5318
                                 14    AE  9F 005E8        PUSHAB  STRING_PTR                                 ; 5324
                                 0C    AE  9F 005EB        PUSHAB  CURRENT_LEN                                ;
                                 4C    AA  9F 005EE        PUSHAB  SHEET_FORMAT                               ;
                       69              03  FB 005F1        CALLS   #3, SYS$FAO                                ;
                       65        08    AE  A0 005F4        ADDW2   CURRENT_LEN, (R5)                          ; 5326
              18       AE        08    AE  C0 005F8        ADDL2   CURRENT_LEN, STRING_PTR+4                  ; 5327
              14       AE              65  3C 005FD        MOVZWL  (R5), STRING_PTR                           ; 5328
  14  AE 00000200      8F        14    AE  C3 00601        SUBL3   STRING_PTR, #512, STRING_PTR               ;
            0200       8F              65  B1 0060B 35$:   CMPW    (R5), #512                                 ; 5331
                                       59  1A 00610        BGTRU   39$                                        ;
  23                                   67  E9 00612        BLBC    (R7), 36$                                  ; 5338
                                 14    AE  9F 00615        PUSHAB  STRING_PTR                                 ; 5344
                                 0C    AE  9F 00618        PUSHAB  CURRENT_LEN                                ;
                                 5C    AA  9F 0061B        PUSHAB  SPACE_FORMAT                               ;
                       69              03  FB 0061E        CALLS   #3, SYS$FAO                                ;
                       65        08    AE  A0 00621        ADDW2   CURRENT_LEN, (R5)                          ; 5346
              18       AE        08    AE  C0 00625        ADDL2   CURRENT_LEN, STRING_PTR+4                  ; 5347
              14       AE              65  3C 0062A        MOVZWL  (R5), STRING_PTR                           ; 5348
  14  AE 00000200      8F        14    AE  C3 0062E        SUBL3   STRING_PTR, #512, STRING_PTR               ;
            0200       8F              65  B1 00638 36$:   CMPW    (R5), #512                                 ; 5351
                                       5B  1A 0063D 37$:   BGTRU   41$                                        ;
  23                   68              02  E1 0063F        BBC     #2, (R8), 38$                              ; 5358
                                 14    AE  9F 00643        PUSHAB  STRING_PTR                                 ; 5364
                                 0C    AE  9F 00646        PUSHAB  CURRENT_LEN                                ;
                                 70    AA  9F 00649        PUSHAB  TRAILER_FORMAT                             ;
                       69              03  FB 0064C        CALLS   #3, SYS$FAO                                ;
                       65        08    AE  A0 0064F        ADDW2   CURRENT_LEN, (R5)                          ; 5366
              18       AE        08    AE  C0 00653        ADDL2   CURRENT_LEN, STRING_PTR+4                  ; 5367
              14       AE              65  3C 00658        MOVZWL  (R5), STRING_PTR                           ; 5368
  14  AE 00000200      8F        14    AE  C3 0065C        SUBL3   STRING_PTR, #512, STRING_PTR               ;
            0200       8F              65  B1 00666 38$:   CMPW    (R5), #512                                 ; 5371
                                       5C  1A 0066B 39$:   BGTRU   42$                                        ;
  24                   67              06  E1 0066D        BBC     #6, (R7), 40$                              ; 5378
                                 14    AE  9F 00671        PUSHAB  STRING_PTR                                 ; 5384
                                 0C    AE  9F 00674        PUSHAB  CURRENT_LEN                                ;
                               0084    CA  9F 00677        PUSHAB  TRUNCATE_FORMAT                            ;
                       69              03  FB 0067B        CALLS   #3, SYS$FAO                                ;
                       65        08    AE  A0 0067E        ADDW2   CURRENT_LEN, (R5)                          ; 5386
```

```
                    18      AE            08     AE   C0 00682          ADDL2   CURRENT_LEN, STRING_PTR+4                     ; 5387
                    14      AE                   65   3C 00687          MOVZWL  (R5), STRING_PTR                              ; 5388
      14   AE 00000200      8F            14     AE   C3 0068B          SUBL3   STRING_PTR, #512, STRING_PTR                  ;
                  0200      8F                   65   B1 00695 40$:     CMPW    (R5), #512                                    ; 5391
                                                 52   1A 0069A 41$:     BGTRU   45$                                           ;
                                        008C     C4   DD 0069C          PUSHL   140(R4)                                       ; 5402
                                          18     AE   9F 006A0          PUSHAB  STRING_PTR                                    ;
                                          10     AE   9F 006A3          PUSHAB  CURRENT_LEN                                   ;
                                        0098     CA   9F 006A6          PUSHAB  WIDTH_FORMAT                                  ;
                            69                   04   FB 006AA          CALLS   #4, SYS$FAO                                   ;
                            65            08     AE   A0 006AD          ADDW2   CURRENT_LEN, (R5)                             ; 5404
                    18      AE            08     AE   C0 006B1          ADDL2   CURRENT_LEN, STRING_PTR+4                     ; 5405
                    14      AE                   65   3C 006B6          MOVZWL  (R5), STRING_PTR                              ; 5406
      14   AE 00000200      8F            14     AE   C3 006BA          SUBL3   STRING_PTR, #512, STRING_PTR                  ;
                  0200      8F                   65   B1 006C4          CMPW    (R5), #512                                    ; 5408
                                                 23   1A 006C9 42$:     BGTRU   45$                                           ;
                                                 67   95 006CB          TSTB    (R7)                                          ; 5415
                                                 11   18 006CD          BGEQ    43$                                           ;
                                          14     AE   9F 006CF          PUSHAB  STRING_PTR                                    ; 5421
                                          0C     AE   9F 006D2          PUSHAB  CURRENT_LEN                                   ;
                                        00A8     CA   9F 006D5          PUSHAB  WRAP_FORMAT                                   ;
                            69                   03   FB 006D9          CALLS   #3, SYS$FAO                                   ;
                            65            08     AE   A0 006DC          ADDW2   CURRENT_LEN, (R5)                             ; 5423
                            12                   65   B1 006E0 43$:     CMPW    (R5), #18                                     ; 5428
                                                 02   1A 006E3          BGTRU   44$                                           ;
                                                 65   B4 006E5          CLRW    (R5)                                          ; 5430
                  0200      8F                   65   B1 006E7 44$:     CMPW    (R5), #512                                    ; 5434
                                                 05   1B 006EC          BLEQU   46$                                           ;
                            65          0200     8F   B0 006EE 45$:     MOVW    #512, (R5)                                    ; 5437
                                                      04 006F3 46$:     RET                                                   ; 5441

; Routine Size:  1780 bytes,    Routine Base:  CODE + 2070
```

```
SEF
V04

20
20

2E
2E

20
20

2E
2E

2D
2D

; R
```

```
  1 %sbttl 'GET_QUEUE_QUALIFIERS - Gets the qualifiers pertaining to queues'
  1 !++
  1 ! Functional Description:
  1 !       This routine returns a string containing the all relevant file qualifier
  1 !       information.
  1 !
  1 ! Formal Parameters:
  1 !       SCB             - Address of the SCB
  1 !       STR_DESC        - Desc of String to Return
  1 !       RET_LEN         - Return length of Desc.
  1 !
  1 ! Implicit Inputs:
  1 !               none
  1 !
  1 ! Implicit Outputs:
  1 !               none
  1 !
  1 ! Returned Value:
  1 !               none
  1 !
  1 ! Side Effects:
  1 !               none
  1 !--
  1 ROUTINE GET_QUEUE_QUALIFIERS (
  1                 SCB             : REF $BBLOCK,          ! SCB
  1                 STR_DESC        : REF VECTOR[2],        ! Output buffer desc
  1                 RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
  1                         )       : NOVALUE =
  2 BEGIN
  2 BIND
P 2     BEGIN_FORMAT = $DESCRIPTOR (
  2         'Queue Qualifiers:'),
  2
P 2     RESET_FORMAT = $DESCRIPTOR (
P 2         'RESET=''',
P 2         '!AS',                                          ! - Reset Module
  2         ''''),
  2
P 2     JOB_RESET_MODULE_FORMAT = $DESCRIPTOR (
  2         ' /SEPARATE=('),                                ! - separation
  2
P 2     INSERTION_FORMAT = $DESCRIPTOR (
  2         '!AC');                                         ! - separation flags
  2
  2
  2 LITERAL
  2     K_MAX_BUFFER_SIZE = 512;
  2
  2 LOCAL
  2     INSERT_FLAG : INITIAL (0),
  2     CURRENT_LEN : INITIAL (0),
  2     STRING_PTR  : VECTOR [2];               ! Pointer to current string
  2
  2 STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE;
  2 STRING_PTR[ADDR] = .STR_DESC[ADDR];         ! init address
  2
  2 RET_LEN[0] = 0;
```

```
; 4586         5499  2
; 4587     P   5500  2  $FAO (   BEGIN_FORMAT,
; 4588     P   5501  2           CURRENT_LEN,                      ! return length
; 4589     P   5502  2           STRING_PTR[0],                    ! address of string
; 4590         5503  2       );
; 4591         5504  2
; 4592         5505  2  RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4593         5506  2  STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4594         5507  2  STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4595         5508  2
; 4596         5509  2
; 4597         5510  2  IF .SEPARATE_FLAG_ (JOB_BURST) OR
; 4598         5511  2      .SEPARATE_FLAG_ (JOB_FLAG) OR
; 4599         5512  2      .SEPARATE_FLAG_ (JOB_TRAILER) OR
; 4600         5513  2      .SCB_SIZE_ (JOB_RESET_MODULES)
; 4601         5514  2  THEN
; 4602         5515  3      BEGIN
; 4603     P   5516  3      $FAO (   JOB_RESET_MODULE_FORMAT,                ! /SEPARATE
; 4604     P   5517  3               CURRENT_LEN,                            ! return length
; 4605     P   5518  3               STRING_PTR[0]                           ! address of string
; 4606         5519  3           );
; 4607         5520  3
; 4608         5521  3      RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4609         5522  3      STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4610         5523  3      STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4611         5524  3
; 4612         5525  3      IF .SEPARATE_FLAG_ (JOB_BURST)
; 4613         5526  3      THEN
; 4614         5527  4          BEGIN
; 4615     P   5528  4          $FAO (
; 4616     P   5529  4              INSERTION_FORMAT,
; 4617     P   5530  4              CURRENT_LEN,                            ! return length
; 4618     P   5531  4              STRING_PTR[0],                          ! address of string
; 4619     P   5532  4              UPLIT BYTE (%ASCIC'BURST')
; 4620         5533  4          );
; 4621         5534  4
; 4622         5535  4          INSERT_FLAG = 1;
; 4623         5536  4          RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4624         5537  4          STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4625         5538  4          STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4626         5539  3          END;
; 4627         5540  3
; 4628         5541  3      IF .SEPARATE_FLAG_ (JOB_FLAG) AND
; 4629         5542  3          .INSERT_FLAG
; 4630         5543  3      THEN
; 4631         5544  4          BEGIN
; 4632     P   5545  4          $FAO (
; 4633     P   5546  4              INSERTION_FORMAT,
; 4634     P   5547  4              CURRENT_LEN,                            ! return length
; 4635     P   5548  4              STRING_PTR[0],                          ! address of string
; 4636     P   5549  4              UPLIT BYTE (%ASCIC',')
; 4637         5550  4          );
; 4638         5551  4
; 4639         5552  4          RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4640         5553  4          STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4641         5554  4          STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4642         5555  3          END;
```

```

        IF .SEPARATE_FLAG_ (JOB_FLAG)
        THEN
            BEGIN
            $FAO (
                INSERTION_FORMAT,
                CURRENT_LEN,                            ! return length
                STRING_PTR[0],                          ! address of string
                UPLIT BYTE (%ASCIC'FLAG')
            );

            INSERT_FLAG = 1;
            RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
            STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
            STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
            END;

        IF .SCB_SIZE_ (JOB_RESET_MODULES) AND
            .INSERT_FLAG
        THEN
            BEGIN
            $FAO (
                INSERTION_FORMAT,
                CURRENT_LEN,                            ! return length
                STRING_PTR[0],                          ! address of string
                UPLIT BYTE (%ASCIC',')
            );

            RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
            STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
            STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
            END;

        IF .SCB_SIZE_(JOB_RESET_MODULES)
        THEN
            BEGIN
            $FAO (
                RESET_FORMAT,
                CURRENT_LEN,                            ! return length
                STRING_PTR[0],                          ! address of string
                SCB[PSM$Q_JOB_RESET_MODULES]
            );

            RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
            STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
            STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
            END;

        IF .SEPARATE_FLAG_ (JOB_TRAILER) AND
            .INSERT_FLAG
        THEN
            BEGIN
            $FAO (
                INSERTION_FORMAT,
                CURRENT_LEN,                            ! return length
                STRING_PTR[0],                          ! address of string
                UPLIT BYTE (%ASCIC',')
```

```
; 4700          5613  4             );
; 4701          5614  4
; 4702          5615  4             RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4703          5616  4             STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4704          5617  4             STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4705          5618  3             END;
; 4706          5619  3
; 4707          5620  3         IF .SEPARATE_FLAG_ (JOB_TRAILER)
; 4708          5621  3         THEN
; 4709          5622  4             BEGIN
; 4710        P 5623  4             $FAO (
; 4711        P 5624  4                 INSERTION_FORMAT,
; 4712        P 5625  4                 CURRENT_LEN,                          ! return length
; 4713        P 5626  4                 STRING_PTR[0],                        ! address of string
; 4714        P 5627  4                 UPLIT BYTE (%ASCIC'TRAILER')
; 4715          5628  4             );
; 4716          5629  4
; 4717          5630  4             RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4718          5631  4             STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4719          5632  4             STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4720          5633  3             END;
; 4721          5634  3
; 4722        P 5635  3         $FAO (
; 4723        P 5636  3             INSERTION_FORMAT,
; 4724        P 5637  3             CURRENT_LEN,                      ! return length
; 4725        P 5638  3             STRING_PTR[0],                    ! address of string
; 4726        P 5639  3             UPLIT BYTE (%ASCIC')')
; 4727          5640  3             );
; 4728          5641  3
; 4729          5642  3         RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4730          5643  3         STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4731          5644  3         STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4732          5645  2         END;
; 4733          5646  2
; 4734          5647  2   ! Don't print anything if no flags were set
; 4735          5648  2   !
; 4736          5649  2   IF .RET_LEN[0] LEQ 18
; 4737          5650  2   THEN
; 4738          5651  2       RET_LEN[0] = 0;
; 4739          5652  2
; 4740          5653  2   ! Length returned must be less than max string size
; 4741          5654  2   IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4742          5655  2   THEN
; 4743          5656  3       BEGIN
; 4744          5657  3           RET_LEN[0] = 512;
; 4745          5658  3           RETURN;
; 4746          5659  2       END;
; 4747          5660  2
; 4748          5661  1   END;


72  65  69  66  69  6C  61  75  51  20  65  75  65  75  51  02764 P.AFP:   .ASCII  \Queue Qualifiers:\                    ;
                                                      3A  73  02773                                                          ;
                                                              02775            .BLKB   3
                                                    00000011  02778 P.AFO:   .LONG   17                                      ;
                                                    00000000' 0277C            .ADDRESS P.AFP                                ;
```

```
                 22  3D  54  45  53  45  52  02780 P.AFR:   .ASCII   \RESET="\                                         ;
                                 53  41  21  02787          .ASCII   \!AS\                                             ;
                                         22  0278A          .ASCII   \"\                                               ;
                                             0278B          .BLKB    1
                                  0000000B   0278C P.AFQ:   .LONG    11                                                ;
                                  00000000'  02790          .ADDRESS P.AFR
 28  3D  45  54  41  52  41  50  45  53  2F  20  02794 P.AFT:   .ASCII   \ /SEPARATE=(\                                ;
                                  0000000C   027A0 P.AFS:   .LONG    12                                                ;
                                  00000000'  027A4          .ADDRESS P.AFT                                             ;
                                 43  41  21  027A8 P.AFV:   .ASCII   \!AC\                                             ;
                                             027AB          .BLKB    1
                                  00000003   027AC P.AFU:   .LONG    3                                                 ;
                                  00000000'  027B0          .ADDRESS P.AFV                                             ;
                     54  53  52  55  42  05  027B4 P.AFW:   .ASCII   <5>\BURST\                                        ;
                                     2C  01  027BA P.AFX:   .ASCII   <1>\,\                                            ;
                         47  41  4C  46  04  027BC P.AFY:   .ASCII   <4>\FLAG\                                         ;
                                     2C  01  027C1 P.AFZ:   .ASCII   <1>\,\                                            ;
                                     2C  01  027C3 P.AGA:   .ASCII   <1>\,\                                            ;
                 52  45  4C  49  41  52  54  07  027C5 P.AGB:   .ASCII   <7>\TRAILER\                                  ;
                                     29  01  027CD P.AGC:   .ASCII   <1>\)\                                            ;

                                                   BEGIN_FORMAT=            P.AFO
                                                   RESET_FORMAT=            P.AFQ
                                                   JOB_RESET_MODULE_FORMAT=
                                                                            P.AFS
                                                   INSERTION_FORMAT=        P.AFU


                                       00FC 00000 GET_QUEUE_QUALIFIERS:
                                                            .WORD    Save R2,R3,R4,R5,R6,R7                       ; 5465
               57          D8    AF    9E 00002             MOVAB    INSERTION_FORMAT, R7                         ;
               56 00000000G       00    9E 00006             MOVAB    SYS$FAO, R6                                  ;
               5E                 08    C2 0000D             SUBL2    #8, SP                                       ;
                                  55    D4 00010             CLRL     INSERT_FLAG                                  ; 5470
                                  7E    D4 00012             CLRL     CURRENT_LEN                                  ;
          04   AE        0200    8F    3C 00014             MOVZWL   #512, STRING_PTR                             ; 5495
               50          08    AC    D0 0001A             MOVL     STR_DESC, R0                                 ; 5496
          08   AE          04    A0    D0 0001E             MOVL     4(R0), STRING_PTR+4                          ;
               52          0C    AC    D0 00023             MOVL     RET_LEN, R2                                  ; 5498
                                  62    B4 00027             CLRW     (R2)                                         ;
                           04    AE    9F 00029             PUSHAB   STRING_PTR                                   ; 5503
                           04    AE    9F 0002C             PUSHAB   CURRENT_LEN                                  ;
                           CC    A7    9F 0002F             PUSHAB   BEGIN_FORMAT                                 ;
               66                 03    FB 00032             CALLS    #3, SYS$FAO                                  ;
               62                 6E    A0 00035             ADDW2    CURRENT_LEN, (R2)                            ; 5505
          08   AE                 6E    C0 00038             ADDL2    CURRENT_LEN, STRING_PTR+4                    ; 5506
          04   AE                 62    3C 0003C             MOVZWL   (R2), STRING_PTR                             ; 5507
 04   AE 00000200    8F          04    AE    C3 00040             SUBL3    STRING_PTR, #512, STRING_PTR            ;
               53          04    AC    D0 0004A             MOVL     SCB, R3                                      ; 5510
               54        0154    C3    9E 0004E             MOVAB    340(R3), R4                                  ;
     10        64                 05    E0 00053             BBS      #5, (R4), 1$                                 ;
     0C        64                 04    E0 00057             BBS      #4, (R4), 1$                                 ; 5511
               08          01    A4    E8 0005B             BLBS     1(R4), 1$                                    ; 5512
               03        00B0    C3    E8 0005F             BLBS     176(R3), 1$                                  ; 5513
                        0168    31 00064             BRW      9$                                           ;
                           04    AE    9F 00067 1$:         PUSHAB   STRING_PTR                                   ; 5519
                           04    AE    9F 0006A             PUSHAB   CURRENT_LEN                                  ;
```

```
                                 F4    A7  9F 0006D         PUSHAB  JOB_RESET_MODULE_FORMAT                ;
                     66                03  FB 00070         CALLS   #3, SYS$FAO                            ;
                     62                6E  A0 00073         ADDW2   CURRENT_LEN, (R2)                      ; 5521
            08       AE                6E  C0 00076         ADDL2   CURRENT_LEN, STRING_PTR+4              ; 5522
            04       AE                62  3C 0007A         MOVZWL  (R2), STRING_PTR                       ; 5523
  04   AE 00000200   8F          04    AE  C3 0007E         SUBL3   STRING_PTR, #512, STRING_PTR           ;
       26            64                05  E1 00088         BBC     #5, (R4), 2$                           ; 5525
                                 08    A7  9F 0008C         PUSHAB  P.AFW                                  ; 5533
                                 08    AE  9F 0008F         PUSHAB  STRING_PTR                             ;
                                 08    AE  9F 00092         PUSHAB  CURRENT_LEN                            ;
                                       57  DD 00095         PUSHL   R7                                     ;
                     66                04  FB 00097         CALLS   #4, SYS$FAO                            ;
                     55                01  D0 0009A         MOVL    #1, INSERT_FLAG                        ; 5535
                     62                6E  A0 0009D         ADDW2   CURRENT_LEN, (R2)                      ; 5536
            08       AE                6E  C0 000A0         ADDL2   CURRENT_LEN, STRING_PTR+4              ; 5537
            04       AE                62  3C 000A4         MOVZWL  (R2), STRING_PTR                       ; 5538
  04   AE 00000200   8F          04    AE  C3 000A8         SUBL3   STRING_PTR, #512, STRING_PTR           ;
       50            64                04  E1 000B2  2$:    BBC     #4, (R4), 4$                           ; 5541
       23            55                    E9 000B6         BLBC    INSERT_FLAG, 3$                        ; 5542
                                 0E    A7  9F 000B9         PUSHAB  P.AFX                                  ; 5550
                                 08    AE  9F 000BC         PUSHAB  STRING_PTR                             ;
                                 08    AE  9F 000BF         PUSHAB  CURRENT_LEN                            ;
                                       57  DD 000C2         PUSHL   R7                                     ;
                     66                04  FB 000C4         CALLS   #4, SYS$FAO                            ;
                     62                6E  A0 000C7         ADDW2   CURRENT_LEN, (R2)                      ; 5552
            08       AE                6E  C0 000CA         ADDL2   CURRENT_LEN, STRING_PTR+4              ; 5553
            04       AE                62  3C 000CE         MOVZWL  (R2), STRING_PTR                       ; 5554
  04   AE 00000200   8F          04    AE  C3 000D2         SUBL3   STRING_PTR, #512, STRING_PTR           ;
       26            64                04  E1 000DC  3$:    BBC     #4, (R4), 4$                           ; 5557
                                 10    A7  9F 000E0         PUSHAB  P.AFY                                  ; 5565
                                 08    AE  9F 000E3         PUSHAB  STRING_PTR                             ;
                                 08    AE  9F 000E6         PUSHAB  CURRENT_LEN                            ;
                                       57  DD 000E9         PUSHL   R7                                     ;
                     66                04  FB 000EB         CALLS   #4, SYS$FAO                            ;
                     55                01  D0 000EE         MOVL    #1, INSERT_FLAG                        ; 5567
                     62                6E  A0 000F1         ADDW2   CURRENT_LEN, (R2)                      ; 5568
            08       AE                6E  C0 000F4         ADDL2   CURRENT_LEN, STRING_PTR+4              ; 5569
            04       AE                62  3C 000F8         MOVZWL  (R2), STRING_PTR                       ; 5570
  04   AE 00000200   8F          04    AE  C3 000FC         SUBL3   STRING_PTR, #512, STRING_PTR           ;
                     50        00B0    C3  E9 00106  4$:    BLBC    176(R3), 6$                            ; 5573
                     23            55      E9 0010B         BLBC    INSERT_FLAG, 5$                        ; 5574
                                 15    A7  9F 0010E         PUSHAB  P.AFZ                                  ; 5582
                                 08    AE  9F 00111         PUSHAB  STRING_PTR                             ;
                                 08    AE  9F 00114         PUSHAB  CURRENT_LEN                            ;
                                       57  DD 00117         PUSHL   R7                                     ;
                     66                04  FB 00119         CALLS   #4, SYS$FAO                            ;
                     62                6E  A0 0011C         ADDW2   CURRENT_LEN, (R2)                      ; 5584
            08       AE                6E  C0 0011F         ADDL2   CURRENT_LEN, STRING_PTR+4              ; 5585
            04       AE                62  3C 00123         MOVZWL  (R2), STRING_PTR                       ; 5586
  04   AE 00000200   8F          04    AE  C3 00127         SUBL3   STRING_PTR, #512, STRING_PTR           ;
                     25        00B0    C3  E9 00131  5$:    BLBC    176(R3), 6$                            ; 5589
                               00B0    C3  9F 00136         PUSHAB  176(R3)                                ; 5597
                                 08    AE  9F 0013A         PUSHAB  STRING_PTR                             ;
                                 08    AE  9F 0013D         PUSHAB  CURRENT_LEN                            ;
                                 E0    A7  9F 00140         PUSHAB  RESET_FORMAT                           ;
                     66                04  FB 00143         CALLS   #4, SYS$FAO                            ;
                     62                6E  A0 00146         ADDW2   CURRENT_LEN, (R2)                      ; 5599
```

```
                   08  AE              6E  C0 00149         ADDL2   CURRENT_LEN, STRING_PTR+4       ; 5600
                   04  AE              62  3C 0014D         MOVZWL  (R2), STRING_PTR                ; 5601
      04  AE 00000200  8F         04   AE  C3 00151         SUBL3   STRING_PTR, #512, STRING_PTR    ;
                       4D         01   A4  E9 0015B 6$:     BLBC    1(R4), 8$                       ; 5604
                       23              55  E9 0015F         BLBC    INSERT_FLAG, 7$                 ; 5605
                                  17   A7  9F 00162         PUSHAB  P.AGA                           ; 5613
                                  08   AE  9F 00165         PUSHAB  STRING_PTR                      ;
                                  08   AE  9F 00168         PUSHAB  CURRENT_LEN                     ;
                                       57  DD 0016B         PUSHL   R7                              ;
                       66              04  FB 0016D         CALLS   #4, SYS$FAO                     ;
                       62              6E  A0 00170         ADDW2   CURRENT_LEN, (R2)               ; 5615
                   08  AE              6E  C0 00173         ADDL2   CURRENT_LEN, STRING_PTR+4       ; 5616
                   04  AE              62  3C 00177         MOVZWL  (R2), STRING_PTR                ; 5617
      04  AE 00000200  8F         04   AE  C3 0017B         SUBL3   STRING_PTR, #512, STRING_PTR    ;
                       23         01   A4  E9 00185 7$:     BLBC    1(R4), 8$                       ; 5620
                                  19   A7  9F 00189         PUSHAB  P.AGB                           ; 5628
                                  08   AE  9F 0018C         PUSHAB  STRING_PTR                      ;
                                  08   AE  9F 0018F         PUSHAB  CURRENT_LEN                     ;
                                       57  DD 00192         PUSHL   R7                              ;
                       66              04  FB 00194         CALLS   #4, SYS$FAO                     ;
                       62              6E  A0 00197         ADDW2   CURRENT_LEN, (R2)               ; 5630
                   08  AE              6E  C0 0019A         ADDL2   CURRENT_LEN, STRING_PTR+4       ; 5631
                   04  AE              62  3C 0019E         MOVZWL  (R2), STRING_PTR                ; 5632
      04  AE 00000200  8F         04   AE  C3 001A2         SUBL3   STRING_PTR, #512, STRING_PTR    ;
                                  21   A7  9F 001AC 8$:     PUSHAB  P.AGC                           ; 5640
                                  08   AE  9F 001AF         PUSHAB  STRING_PTR                      ;
                                  08   AE  9F 001B2         PUSHAB  CURRENT_LEN                     ;
                                       57  DD 001B5         PUSHL   R7                              ;
                       66              04  FB 001B7         CALLS   #4, SYS$FAO                     ;
                       62              6E  A0 001BA         ADDW2   CURRENT_LEN, (R2)               ; 5642
                   08  AE              6E  C0 001BD         ADDL2   CURRENT_LEN, STRING_PTR+4       ; 5643
                   04  AE              62  3C 001C1         MOVZWL  (R2), STRING_PTR                ; 5644
      04  AE 00000200  8F         04   AE  C3 001C5         SUBL3   STRING_PTR, #512, STRING_PTR    ;
                       12              62  B1 001CF 9$:     CMPW    (R2), #18                       ; 5649
                                       02  1A 001D2         BGTRU   10$                             ;
                                       62  B4 001D4         CLRW    (R2)                            ; 5651
                 0200  8F              62  B1 001D6 10$:    CMPW    (R2), #512                      ; 5654
                                       05  1B 001DB         BLEQU   11$                             ;
                       62       0200   8F  B0 001DD         MOVW    #512, (R2)                      ; 5657
                                           04 001E2 11$:    RET                                     ; 5661

; Routine Size:  483 bytes,    Routine Base:  CODE + 27CF
; 4750          5662   1 %sbttl 'GET_FORM_QUALIFIERS - Get the qualifiers which pertain to forms'
; 4751          5663   1 !++
; 4752          5664   1 ! Functional Description:
; 4753          5665   1 !        This routine returns a string containing the all relevant file qualifier
; 4754          5666   1 !        information.
; 4755          5667   1 !
; 4756          5668   1 ! Formal Parameters:
; 4757          5669   1 !        SCB                 - Address of the SCB
; 4758          5670   1 !        STR_DESC            - Desc of String to Return
; 4759          5671   1 !        RET_LEN             - Return length of Desc.
; 4760          5672   1 !
; 4761          5673   1 ! Implicit Inputs:
; 4762          5674   1 !                 none
; 4763          5675   1 !
; 4764          5676   1 ! Implicit Outputs:
; 4765          5677   1 !                 none
; 4766          5678   1 !
; 4767          5679   1 ! Returned Value:
; 4768          5680   1 !                 none
; 4769          5681   1 !
; 4770          5682   1 ! Side Effects:
; 4771          5683   1 !                 none
; 4772          5684   1 !--
; 4773          5685   1 ROUTINE GET_FORM_QUALIFIERS (
; 4774          5686   1                 SCB                 : REF $BBLOCK,            ! SCB
; 4775          5687   1                 STR_DESC            : REF VECTOR[2],          ! Output buffer desc
; 4776          5688   1                 RET_LEN             : REF VECTOR [,WORD]      ! Return length (word)
; 4777          5689   1                             )       : NOVALUE =
; 4778          5690   2 BEGIN
; 4779          5691   2 BIND
; 4780        P 5692   2     BEGIN_FORMAT = $DESCRIPTOR(
; 4781          5693   2         'Form Qualifiers:');
; 4782          5694   2
; 4783          5695   2 LITERAL
; 4784          5696   2     K_MAX_BUFFER_SIZE = 512;
; 4785          5697   2
; 4786          5698   2 LOCAL
; 4787          5699   2     CURRENT_LEN : INITIAL (0),
; 4788          5700   2     STRING_PTR  : VECTOR [2];               ! Pointer to current string
; 4789          5701   2
; 4790          5702   2
; 4791          5703   2 STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE;
; 4792          5704   2 STRING_PTR[ADDR] = .STR_DESC[ADDR];         ! init address
; 4793          5705   2
; 4794          5706   2 RET_LEN[0] = 0;
; 4795          5707   2
; 4796        P 5708   2 $FAO (  BEGIN_FORMAT,
; 4797        P 5709   2         CURRENT_LEN,                        ! return length
; 4798        P 5710   2         STRING_PTR[0],                      ! address of string
; 4799          5711   2     );
; 4800          5712   2
; 4801          5713   2 RET_LEN[0] = .RET_LEN[0] + .CURRENT_LEN;
; 4802          5714   2 STRING_PTR[ADDR] = .STRING_PTR[ADDR] + .CURRENT_LEN;
; 4803          5715   2 STRING_PTR[SIZE] = K_MAX_BUFFER_SIZE - .RET_LEN[0];
; 4804          5716   2
; 4805          5717   2
; 4806          5718   2
; 4807          5719  2  ! Don't print anything if no flags were set
; 4808          5720  2  !
; 4809          5721  2  IF .RET_LEN[0] LEQ 18
; 4810          5722  2  THEN
; 4811          5723  2      RET_LEN[0] = 0;
; 4812          5724  2
; 4813          5725  2  ! Length returned must be less than max string size
; 4814          5726  2  IF .RET_LEN[0] GTR K_MAX_BUFFER_SIZE
; 4815          5727  2  THEN
; 4816          5728  3      BEGIN
; 4817          5729  3          RET_LEN[0] = 512;
; 4818          5730  3          RETURN;
; 4819          5731  2      END;
; 4820          5732  2
; 4821          5733  1  END;


73  72  65  69  66  69  6C  61  75  51  20  6D  72  6F  46 029B2 P.AGE:  .ASCII  \Form Qualifiers:\                            ;
                                                        3A 029C1                                                               ;
                                                           029C2         .BLKB   2
                                                  00000010 029C4 P.AGD:  .LONG   16                                            ;
                                                 00000000' 029C8         .ADDRESS P.AGE                                        ;

                                                                 BEGIN_FORMAT=       P.AGD


                                                      0004 00000 GET_FORM_QUALIFIERS:
                                                                         .WORD   Save R2                                       ; 5685
                                        5E           08  C2 00002         SUBL2   #8, SP                                        ; 5690
                                                     7E  D4 00005         CLRL    CURRENT_LEN                                   ; 5690
                                  04    AE    0200   8F  3C 00007         MOVZWL  #512, STRING_PTR                              ; 5703
                                        50      08   AC  D0 0000D         MOVL    STR_DESC, R0                                  ; 5704
                                  08    AE      04   A0  D0 00011         MOVL    4(R0), STRING_PTR+4
                                        52      0C   AC  D0 00016         MOVL    RET_LEN, R2                                   ; 5706
                                                     62  B4 0001A         CLRW    (R2)
                                                04   AE  9F 0001C         PUSHAB  STRING_PTR                                    ; 5711
                                                04   AE  9F 0001F         PUSHAB  CURRENT_LEN
                                                D3   AF  9F 00022         PUSHAB  BEGIN_FORMAT
                             00000000G  00           03  FB 00025         CALLS   #3, SYS$FAO
                                        62           6E  A0 0002C         ADDW2   CURRENT_LEN, (R2)                             ; 5713
                                  08    AE           6E  C0 0002F         ADDL2   CURRENT_LEN, STRING_PTR+4                     ; 5714
                                  04    AE           62  3C 00033         MOVZWL  (R2), STRING_PTR                              ; 5715
                    04   AE 00000200    8F      04   AE  C3 00037         SUBL3   STRING_PTR, #512, STRING_PTR
                                        12           62  B1 00041         CMPW    (R2), #18                                     ; 5721
                                                     02  1A 00044         BGTRU   1$
                                                     62  B4 00046         CLRW    (R2)                                          ; 5723
                                0200    8F           62  B1 00048 1$:     CMPW    (R2), #512                                    ; 5726
                                                     05  1B 0004D         BLEQU   2$
                                        62    0200   8F  B0 0004F         MOVW    #512, (R2)                                    ; 5729
                                                         04 00054 2$:     RET                                                   ; 5733

; Routine Size:  85 bytes,    Routine Base:  CODE + 29CC
; 4823          5734  1 %sbttl 'GET_USER_NOTE- Insert a Note into the Page'
; 4824          5735  1 !++
; 4825          5736  1 ! Functional Description:
; 4826          5737  1 !       This routine gets a note as specified by the user for the frame.
; 4827          5738  1 !
; 4828          5739  1 ! Formal Parameters:
; 4829          5740  1 !       SCB             - Address of the SCB
; 4830          5741  1 !       STR_DESC        - Desc of String to Return
; 4831          5742  1 !       RET_LEN         - Return length of Desc.
; 4832          5743  1 !
; 4833          5744  1 ! Implicit Inputs:
; 4834          5745  1 !                       none
; 4835          5746  1 !
; 4836          5747  1 ! Implicit Outputs:
; 4837          5748  1 !                       none
; 4838          5749  1 !
; 4839          5750  1 ! Returned Value:
; 4840          5751  1 !                       none
; 4841          5752  1 !
; 4842          5753  1 ! Side Effects:
; 4843          5754  1 !                       none
; 4844          5755  1 !--
; 4845          5756  1 ROUTINE GET_USER_NOTE (
; 4846          5757  1                 SCB             : REF $BBLOCK,          ! SCB
; 4847          5758  1                 STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 4848          5759  1                 RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 4849          5760  1                         )       : NOVALUE =
; 4850          5761  2 BEGIN
; 4851          5762  2 BIND
; 4852        P 5763  2     NOTE_FULL_FORMAT = $DESCRIPTOR (
; 4853          5764  2         'NOTE: !AF'                     );              ! - user note
; 4854          5765  2 LITERAL
; 4855          5766  2     K_MIN_NOTE_LEN = 6;
; 4856          5767  2 LOCAL
; 4857          5768  2     STATUS;
; 4858          5769  2
; 4859        P 5770  2 STATUS = $FAO ( NOTE_FULL_FORMAT,
; 4860        P 5771  2                 RET_LEN[0],
; 4861        P 5772  2                 STR_DESC[0],
; 4862        P 5773  2                 .SCB_SIZE_ (NOTE),
; 4863          5774  2                 .SCB_ADDR_ (NOTE));     ! user note
; 4864          5775  2
; 4865          5776  2 IF .RET_LEN[0] LEQ K_MIN_NOTE_LEN                     ! print nothing... no note
; 4866          5777  2     THEN RET_LEN[0] = 0;
; 4867          5778  2
; 4868          5779  2
; 4869          5780  2 RETURN SS$_NORMAL;
; 4870          5781  1 END;


                46  41  21  20  3A  45  54  4F  4E  02A21 P.AGG:  .ASCII  \NOTE: !AF\                                ;
                                                    02A2A         .BLKB   2
                                          00000009  02A2C P.AGF:  .LONG   9                                          ;
                                          00000000' 02A30         .ADDRESS P.AGG                                     ;

                                                        NOTE_FULL_FORMAT=   P.AGF
```

```
                                 0000 00000 GET_USER_NOTE:
                                                  .WORD   Save nothing                         ; 5756
             50         04   AC   D0 00002        MOVL    SCB, R0                              ; 5774
                      00D8   C0   DD 00006        PUSHL   216(R0)                              ;
             7E       00D4   C0   3C 0000A        MOVZWL  212(R0), -(SP)                       ;
                        08   AC   DD 0000F        PUSHL   STR_DESC                             ;
                        0C   AC   DD 00012        PUSHL   RET_LEN                              ;
                        E0   AF   9F 00015        PUSHAB  NOTE_FULL_FORMAT                     ;
00000000G    00              05   FB 00018        CALLS   #5, SYS$FAO                          ;
             06         0C   BC   B1 0001F        CMPW    @RET_LEN, #6                         ; 5776
                             03   1A 00023        BGTRU   1$                                   ;
                        0C   BC   B4 00025        CLRW    @RET_LEN                             ; 5777
                                  04 00028 1$:    RET                                          ; 5781

; Routine Size:  41 bytes,    Routine Base:  CODE + 2A34
; 4872          5782  1 %sbttl 'GET_RECEIPT_BOX - Insert a "Received Box" into the Page'
; 4873          5783  1 !++
; 4874          5784  1 ! Functional Description:
; 4875          5785  1 !       This routine gets a note as specified by the user for the frame.
; 4876          5786  1 !
; 4877          5787  1 ! Formal Parameters:
; 4878          5788  1 !       SCB             - Address of the SCB
; 4879          5789  1 !       STR_DESC        - Desc of String to Return
; 4880          5790  1 !       RET_LEN         - Return length of Desc.
; 4881          5791  1 !
; 4882          5792  1 ! Implicit Inputs:
; 4883          5793  1 !               none
; 4884          5794  1 !
; 4885          5795  1 ! Implicit Outputs:
; 4886          5796  1 !               none
; 4887          5797  1 !
; 4888          5798  1 ! Returned Value:
; 4889          5799  1 !               none
; 4890          5800  1 !
; 4891          5801  1 ! Side Effects:
; 4892          5802  1 !               none
; 4893          5803  1 !--
; 4894          5804  1 ROUTINE GET_RECEIPT_BOX (
; 4895          5805  1                 SCB             : REF $BBLOCK,          ! SCB
; 4896          5806  1                 STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 4897          5807  1                 RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 4898          5808  1                         )       : NOVALUE =
; 4899          5809  2 BEGIN
; 4900          5810  2 BIND
; 4901        P 5811  2     NOTE132_FORMAT = $DESCRIPTOR (
; 4902        P 5812  2             '+--------------------------------------+',
; 4903        P 5813  2             '!                                      !',
; 4904        P 5814  2             '! Received: .......................... !',
; 4905        P 5815  2             '!                                      !',
; 4906        P 5816  2             '! Date    : .......................... !',
; 4907        P 5817  2             '!                                      !',
; 4908        P 5818  2             '! Operator: .......................... !',
; 4909        P 5819  2             '+--------------------------------------+'
; 4910          5820  2                     ) : VECTOR;     ! - receipt box
; 4911          5821  2
; 4912          5822  2 CH$MOVE (.NOTE132_FORMAT[0], .NOTE132_FORMAT[1], .STR_DESC[ADDR]);
; 4913          5823  2 RET_LEN[0] = .NOTE132_FORMAT[0];
; 4914          5824  2
; 4915          5825  2 RETURN SS$_NORMAL;
; 4916          5826  1 END;


2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2B  02A5D P.AGI:  .ASCII  \+--------------------------------------+\  ;
2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D  02A6C                                                            ;
               2B 2D 2D 2D 2D 2D 2D 2D 2D 2D  02A7B                                                            ;
20 20 20 20 20 20 20 20 20 20 20 20 20 20 21  02A85          .ASCII  \!                                      !\  ;
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20  02A94                                                            ;
               21 20 20 20 20 20 20 20 20 20  02AA3                                                            ;
2E 2E 2E 20 3A 64 65 76 69 65 63 65 52 20 21  02AAD          .ASCII  \! Received: .......................... !\  ;
2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E 2E  02ABC                                                            ;
               21 20 2E 2E 2E 2E 2E 2E 2E 2E  02ACB                                                            ;
```

```
20  20  20  20  20  20  20  20  20  20  20  20  20  20  21  02AD5            .ASCII  \!                                       !\   ;
20  20  20  20  20  20  20  20  20  20  20  20  20  20  20  02AE4                                                                 ;
                    21  20  20  20  20  20  20  20  20  20  02AF3                                                                 ;
2E  2E  2E  20  3A  20  20  20  20  65  74  61  44  20  21  02AFD            .ASCII  \! Date     : ......................... !\   ;
2E  2E  2E  2E  2E  2E  2E  2F  2E  2E  2E  2E  2E  2E  2E  02B0C                                                                 ;
                    21  20  2E  2E  2E  2E  2E  2E  2E  2E  02B1B                                                                 ;
20  20  20  20  20  20  20  20  20  20  20  20  20  20  21  02B25            .ASCII  \!                                       !\   ;
20  20  20  20  20  20  20  20  20  20  20  20  20  20  20  02B34                                                                 ;
                    21  20  20  20  20  20  20  20  20  20  02B43                                                                 ;
2E  2E  2E  20  3A  72  6F  74  61  72  65  70  4F  20  21  02B4D            .ASCII  \! Operator: ......................... !\   ;
2E  2E  2E  2E  2E  2E  2E  2E  2E  2E  2E  2E  2E  2E  2E  02B5C                                                                 ;
                    21  20  2E  2E  2E  2E  2E  2E  2E  2E  02B6B                                                                 ;
2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2B  02B75            .ASCII  \+---------------------------------------+\   ;
2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  2D  02B84                                                                 ;
                    2B  2D  2D  2D  2D  2D  2D  2D  2D  2D  02B93                                                                 ;
                                                            02B9D            .BLKB   3
                                                00000140    02BA0  P.AGH:    .LONG   320                                          ;
                                                00000000'   02BA4            .ADDRESS P.AGI                                       ;

                                                                   NOTE132_FORMAT=       P.AGH



                                                        003C 00000 GET_RECEIPT_BOX:
                                                                             .WORD   Save R2,R3,R4,R5                                  ; 5804
                                    50        08   AC   D0 00002             MOVL    STR_DESC, R0                                      ; 5822
                    04   B0    F1   BF        EF   AF   28 00006             MOVC3   NOTE132_FORMAT, @NOTE132_FORMAT+4, @4(R0)         ;
                               0C   BC        E8   AF   B0 0000D             MOVW    NOTE132_FORMAT, @RET_LEN                          ; 5823
                                                        04 00012             RET                                                       ; 5826

; Routine Size:  19 bytes,    Routine Base:  CODE + 2BA8
; 4918          5827  1 %sbttl 'GET_RULER_FINE - Insert a fine "RULER" into the Page'
; 4919          5828  1 !++
; 4920          5829  1 ! Functional Description:
; 4921          5830  1 !       This routine gets a fine ruler '1234567890'
; 4922          5831  1 !
; 4923          5832  1 ! Formal Parameters:
; 4924          5833  1 !       SCB             - Address of the SCB
; 4925          5834  1 !       STR_DESC        - Desc of String to Return
; 4926          5835  1 !       RET_LEN         - Return length of Desc.
; 4927          5836  1 !
; 4928          5837  1 ! Implicit Inputs:
; 4929          5838  1 !               none
; 4930          5839  1 !
; 4931          5840  1 ! Implicit Outputs:
; 4932          5841  1 !               none
; 4933          5842  1 !
; 4934          5843  1 ! Returned Value:
; 4935          5844  1 !               none
; 4936          5845  1 !
; 4937          5846  1 ! Side Effects:
; 4938          5847  1 !               none
; 4939          5848  1 !--
; 4940          5849  1 ROUTINE GET_RULER_FINE (
; 4941          5850  1               SCB             : REF $BBLOCK,          ! SCB
; 4942          5851  1               STR_DESC        : REF VECTOR[2],        ! Output buffer desc
; 4943          5852  1               RET_LEN         : REF VECTOR [,WORD]    ! Return length (word)
; 4944          5853  1                       )       : NOVALUE =
; 4945          5854  2 BEGIN
; 4946          5855  2 BIND
; 4947        P 5856  2     VMS_FORMAT = $DESCRIPTOR (
; 4948        P 5857  2                 '1234567890'
; 4949          5858  2                 );
; 4950          5859  2
; 4951        P 5860  2 $FAO (  VMS_FORMAT,
; 4952        P 5861  2         RET_LEN[0],
; 4953        P 5862  2         STR_DESC[0]
; 4954          5863  2         );
; 4955          5864  2
; 4956          5865  2 RETURN SS$_NORMAL;
; 4957          5866  1 END;


                 30  39  38  37  36  35  34  33  32  31   02BBB P.AGK:   .ASCII  \1234567890\                          ;
                                                          02BC5          .BLKB   3
                                                0000000A  02BC8 P.AGJ:   .LONG   10                                    ;
                                                00000000' 02BCC          .ADDRESS P.AGK                                ;

                                                          VMS_FORMAT=            P.AGJ


                                                  0000 00000 GET_RULER_FINE:
                                                                         .WORD   Save nothing                          ; 5849
                                           08   AC  DD 00002             PUSHL   STR_DESC                              ; 5863
                                           0C   AC  DD 00005             PUSHL   RET_LEN                               ;
                                           ED   AF  9F 00008             PUSHAB  VMS_FORMAT                            ;
                       00000000G  00            03  FB 0000B             CALLS   #3, SYS$FAO                           ;
```

```
                                                 04 00012         RET                                          ; 5866

; Routine Size:  19 bytes,     Routine Base:  CODE + 2BD0
; 4959          5867  1 %sbttl 'GET_RULER_COARSE - Insert a coarse "RULER" into the Page'
; 4960          5868  1 !++
; 4961          5869  1 ! Functional Description:
; 4962          5870  1 !         This routine gets a coarse ruler '1...2...3...' for the frame.
; 4963          5871  1 !
; 4964          5872  1 ! Formal Parameters:
; 4965          5873  1 !       SCB             - Address of the SCB
; 4966          5874  1 !       STR_DESC        - Desc of String to Return
; 4967          5875  1 !       RET_LEN         - Return length of Desc.
; 4968          5876  1 !
; 4969          5877  1 ! Implicit Inputs:
; 4970          5878  1 !                none
; 4971          5879  1 !
; 4972          5880  1 ! Implicit Outputs:
; 4973          5881  1 !                none
; 4974          5882  1 !
; 4975          5883  1 ! Returned Value:
; 4976          5884  1 !                none
; 4977          5885  1 !
; 4978          5886  1 ! Side Effects:
; 4979          5887  1 !                none
; 4980          5888  1 !--
; 4981          5889  1 ROUTINE GET_RULER_COARSE (
; 4982          5890  1                  SCB                : REF $BBLOCK,           ! SCB
; 4983          5891  1                  STR_DESC           : REF VECTOR[2],         ! Output buffer desc
; 4984          5892  1                  RET_LEN            : REF VECTOR [,WORD]     ! Return length (word)
; 4985          5893  1                              )      : NOVALUE =
; 4986          5894  2 BEGIN
; 4987          5895  2 BIND
; 4988        P 5896  2      VMS_FORMAT = $DESCRIPTOR (
; 4989        P 5897  2                  '         1',
; 4990        P 5898  2                  '         2',
; 4991        P 5899  2                  '         3',
; 4992        P 5900  2                  '         4',
; 4993        P 5901  2                  '         5',
; 4994        P 5902  2                  '         6',
; 4995        P 5903  2                  '         7',
; 4996        P 5904  2                  '         8',
; 4997        P 5905  2                  '         9',
; 4998        P 5906  2                  '         0'
; 4999          5907  2                  );
; 5000          5908  2
; 5001        P 5909  2 $FAO (  VMS_FORMAT,
; 5002        P 5910  2         RET_LEN[0],
; 5003        P 5911  2         STR_DESC[0]
; 5004          5912  2         );
; 5005          5913  2
; 5006          5914  2 RETURN SS$_NORMAL;
; 5007          5915  1 END;


                     31  20  20  20  20  20  20  20  20  20  02BE3 P.AGM:  .ASCII  \         1\                           :
                     32  20  20  20  20  20  20  20  20  20  02BED         .ASCII  \         2\                           :
                     33  20  20  20  20  20  20  20  20  20  02BF7         .ASCII  \         3\                           :
                     34  20  20  20  20  20  20  20  20  20  02C01         .ASCII  \         4\                           :
                     35  20  20  20  20  20  20  20  20  20  02C0B         .ASCII  \         5\                           :
```

```
      36  20  20  20  20  20  20  20  20  20  02C15            .ASCII  \        6\                                 ;
      37  20  20  20  20  20  20  20  20  20  02C1F            .ASCII  \        7\                                 ;
      38  20  20  20  20  20  20  20  20  20  02C29            .ASCII  \        8\                                 ;
      39  20  20  20  20  20  20  20  20  20  02C33            .ASCII  \        9\                                 ;
      30  20  20  20  20  20  20  20  20  20  02C3D            .ASCII  \        0\                                 ;
                                              02C47            .BLKB   1
                                   00000064   02C48 P.AGL:     .LONG   100                                          ;
                                   00000000'  02C4C            .ADDRESS P.AGM                                       ;

                                                  VMS_FORMAT=          P.AGL


                                        0000 00000 GET_RULER_COARSE:
                                                           .WORD   Save nothing                                  ; 5889
                                  08   AC  DD 00002        PUSHL   STR_DESC                                      ; 5912
                                  0C   AC  DD 00005        PUSHL   RET_LEN                                       ;
                                  ED   AF  9F 00008        PUSHAB  VMS_FORMAT                                    ;
                   00000000G  00       03  FB 0000B        CALLS   #3, SYS$FAO                                   ;
                                           04 00012        RET                                                   ; 5915

; Routine Size:  19 bytes,    Routine Base:  CODE + 2C50
; 5009          5916  1 %sbttl 'GET_FORM_SIZE - Determine the Size of Form Specified'
; 5010          5917  1 !++
; 5011          5918  1 ! Functional Description:
; 5012          5919  1 !       This routine determines the which standard form is
; 5013          5920  1 !       specified by interrogating the SCB for length and width
; 5014          5921  1 !       Standard forms sizes include: 132x66, 132x51, 80x66, 80x51 or
; 5015          5922  1 !       40xany_length.  Otherwise form size is "non_std".
; 5016          5923  1 !
; 5017          5924  1 ! Formal Parameters:
; 5018          5925  1 !               SCB     - Address of the SCB
; 5019          5926  1 !
; 5020          5927  1 ! Implicit Inputs:
; 5021          5928  1 !               none
; 5022          5929  1 !
; 5023          5930  1 ! Implicit Outputs:
; 5024          5931  1 !               none
; 5025          5932  1 !
; 5026          5933  1 ! Returned Value:
; 5027          5934  1 !               none
; 5028          5935  1 !
; 5029          5936  1 ! Side Effects:
; 5030          5937  1 !               none
; 5031          5938  1 !--
; 5032          5939  1 ROUTINE GET_FORM_SIZE (
; 5033          5940  1                         SCB     : REF $BBLOCK
; 5034          5941  1         ):      NOVALUE =
; 5035          5942  2 BEGIN
; 5036          5943  2
; 5037          5944  2     SCB[PSM$L_PAGE_WIDTH] = .SCB[PSM$L_FORM_WIDTH];
; 5038          5945  2     IF .SCB[PSM$L_PAGE_WIDTH] GTRU 200
; 5039          5946  2     THEN
; 5040          5947  2         SCB[PSM$L_PAGE_WIDTH]  = 200;
; 5041          5948  2
; 5042          5949  2     SCB[PSM$L_PAGE_LENGTH] = .SCB[PSM$L_FORM_LENGTH];
; 5043          5950  2     IF .SCB[PSM$L_PAGE_LENGTH] GTRU 100
; 5044          5951  2     THEN
; 5045          5952  2         SCB[PSM$L_PAGE_LENGTH]  = 100;
; 5046          5953  2
; 5047          5954  2     WHILE .SCB[PSM$L_PAGE_LENGTH] LSSU 40
; 5048          5955  2     DO
; 5049          5956  2         SCB[PSM$L_PAGE_LENGTH]  =
; 5050          5957  2             .SCB[PSM$L_PAGE_LENGTH] + .SCB[PSM$L_FORM_LENGTH];
; 5051          5958  2
; 5052          5959  1 END;



                                                  0000 00000 GET_FORM_SIZE:
                                                                     .WORD   Save nothing                                          ; 5959
                                  50        04    AC   D0 00002      MOVL    SCB, R0                                               ; 5944
                                  51      0200    C0   9E 00006      MOVAB   512(R0), R1                                           ;
                                  61      008C    C0   D0 0000B      MOVL    140(R0), (R1)                                         ;
                        000000C8  8F              61   D1 00010      CMPL    (R1), #200                                            ; 5945
                                                  04   1B 00017      BLEQU   1$                                                    ;
                                  61        C8    8F   9A 00019      MOVZBL  #200, (R1)                                            ; 5947
```

```
                            51       01F8     C0  9E 0001D 1$:      MOVAB    504(R0), R1                                ; 5949
                            61         78     A0  D0 00022          MOVL     120(R0), (R1)                              ;
                  00000064  8F                61  D1 00026          CMPL     (R1), #100                                 ; 5950
                                              04  1B 0002D          BLEQU    2$                                         ;
                            61         64     8F  9A 0002F          MOVZBL   #100, (R1)                                 ; 5952
                            28                61  D1 00033 2$:      CMPL     (R1), #40                                  ; 5954
                                              06  1E 00036          BGEQU    3$                                         ;
                            61         78     A0  C0 00038          ADDL2    120(R0), (R1)                              ; 5957
                                              F5  11 0003C          BRB      2$                                         ; 5956
                                                  04 0003E 3$:      RET                                                 ; 5959

; Routine Size:  63 bytes,     Routine Base:  CODE + 2C63
```

```
1 %sbttl 'FILL_FRAME - Insert Information into this Frame of the Page'
1 !++
1 ! Functional Description:
1 !         This procedure inserts a character into an array(frame)
1 !         until no room left in the frame.
1 !
1 ! Formal Parameters:
1 !         SCB             - Address of the SCB
1 !         CHAR            - Descriptor of String to Insert
1 !         FRAME_PTR       - Address of first byte of Frame
1 !         FRAME_LENGTH    - Length of Frame
1 !         FRAME_WIDTH     - Width of Frame
1 !
1 ! Implicit Inputs:
1 !                 none
1 !
1 ! Implicit Outputs:
1 !                 none
1 !
1 ! Returned Value:
1 !                 none
1 !
1 ! Side Effects:
1 !                 none
1 !--
1 ROUTINE FILL_FRAME (
1         SCB             : REF $BBLOCK,
1         CHAR,
1         FRAME_PTR       : REF PAGE_ARRAY,
1         FRAME_WIDTH,
1         FRAME_LENGTH
1         ):      NOVALUE =
2 BEGIN
2
2 LOCAL   PTR             : REF PAGE_ARRAY,
2         LOC_FRAME_LENGTH,
2         LOC_FRAME_WIDTH ;
2
2 ! Check for dumb calls
2 !
3 IF (.FRAME_LENGTH LEQ 0) OR (.FRAME_WIDTH LEQ 0)
2 THEN
2     RETURN;
2 IF  (.FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]) OR
3     (.FRAME_WIDTH GTR .SCB[PSM$L_PAGE_QIDTH])
2 THEN
2     RETURN;
2
2 ! Check page boundary conditions
2 !
2 LOC_FRAME_LENGTH = .FRAME_LENGTH;
2 IF .LOC_FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]
2 THEN
2     LOC_FRAME_LENGTH = .SCB[PSM$L_PAGE_LENGTH];         ! stay in page bounds
2
2 LOC_FRAME_WIDTH  = .FRAME_WIDTH;
2 IF .LOC_FRAME_WIDTH GTR .SCB[PSM$L_PAGE_WIDTH]
```

```
; 5111          6017   2 THEN
; 5112          6018   2     LOC_FRAME_WIDTH = .SCB[PSM$L_PAGE_WIDTH];            ! stay in page bounds
; 5113          6019   2
; 5114          6020   2
; 5115          6021   2 PTR = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]];
; 5116          6022   2
; 5117          6023   2
; 5118          6024   2 DECR L FROM .LOC_FRAME_LENGTH TO 1 DO
; 5119          6025   3     BEGIN
; 5120          6026   3     CH$FILL( .CHAR, .LOC_FRAME_WIDTH,.PTR);
; 5121          6027   3
; 5122          6028   3     PTR = .PTR + .SCB[PSM$L_PAGE_WIDTH];                 ! Address calc. is based
; 5123          6029   3                                                          !  on Form Width
; 5124          6030   2     END;
; 5125          6031   1 END;



                                                        03FC 00000 FILL_FRAME:
                                                                           .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9               ; 5985
                                       52        14   AC  D0 00002         MOVL    FRAME_LENGTH, R2                            ; 6000
                                                      4C  15 00006         BLEQ    5$                                          ;
                                                 10   AC  D5 00008         TSTL    FRAME_WIDTH                                 ;
                                                      47  15 0000B         BLEQ    5$                                          ;
                                       50        04   AC  D0 0000D         MOVL    SCB, R0                                     ; 6003
                                       51      01F8   C0  D0 00011         MOVL    504(R0), R1                                 ;
                                       51             52  D1 00016         CMPL    R2, R1                                      ;
                                                      39  14 00019         BGTR    5$                                          ;
                                 0200  C0        10   AC  D1 0001B         CMPL    FRAME_WIDTH, 512(R0)                        ; 6004
                                                      31  14 00021         BGTR    5$                                          ;
                                       56             52  D0 00023         MOVL    R2, LOC_FRAME_LENGTH                        ; 6010
                                       51             56  D1 00026         CMPL    LOC_FRAME_LENGTH, R1                        ; 6011
                                                      03  15 00029         BLEQ    1$                                          ;
                                       56             51  D0 0002B         MOVL    R1, LOC_FRAME_LENGTH                        ; 6013
                                       59        10   AC  D0 0002E 1$:     MOVL    FRAME_WIDTH, LOC_FRAME_WIDTH                ; 6015
                                       58      0200   C0  D0 00032         MOVL    512(R0), R8                                 ; 6016
                                       58             59  D1 00037         CMPL    LOC_FRAME_WIDTH, R8                         ;
                                                      03  15 0003A         BLEQ    2$                                          ;
                                       59             58  D0 0003C         MOVL    R8, LOC_FRAME_WIDTH                         ; 6018
                                       57        0C   AC  D0 0003F 2$:     MOVL    FRAME_PTR, PTR                              ; 6021
                                                      56  D6 00043         INCL    L                                           ; 6026
                                                      0A  11 00045         BRB     4$                                          ;
           59       08   AC            6E             00  2C 00047 3$:     MOVC5   #0, (SP), CHAR, LOC_FRAME_WIDTH, (PTR)      ;
                                                      67     0004D                                                             ;
                                       57             58  C0 0004E         ADDL2   R8, PTR                                     ; 6028
                                       F3             56  F5 00051 4$:     SOBGTR  L, 3$                                       ; 6024
                                                          04 00054 5$:     RET                                                 ; 6031

; Routine Size:  85 bytes,    Routine Base:  CODE + 2CA2
```

```
1 %sbttl 'SCROLL_FRAME - Insert Information into this Frame of the Page'
1 !++
1 ! Functional Description:
1 !         This procedure inserts a string into an array(frame) repeatedly
1 !         until no room is left in the frame.
1 !
1 ! Formal Parameters:
1 !         SCB             - Address of the SCB
1 !         CHAR_STRING     - Descriptor of String to Insert
1 !         FRAME_PTR       - Address of first byte of Frame
1 !         FRAME_LENGTH    - Length of Frame
1 !         FRAME_WIDTH     - Width of Frame
1 !
1 ! Implicit Inputs:
1 !                 none
1 !
1 ! Implicit Outputs:
1 !                 none
1 !
1 ! Returned Value:
1 !                 none
1 !
1 ! Side Effects:
1 !                 none
1 !--
1 ROUTINE SCROLL_FRAME (
1         SCB             : REF $BBLOCK,
1         CHAR_STRING     : REF VECTOR[2],
1         FRAME_PTR       : REF PAGE_ARRAY,
1         FRAME_WIDTH     ,
1         FRAME_LENGTH
1         )  :    NOVALUE =
2 BEGIN
2
2 LOCAL   PTR             : REF PAGE_ARRAY,
2         LOC_FRAME_LENGTH,
2         LOC_FRAME_WIDTH ,
2         TEMP_PTR        ,
2         START_CNT       : INITIAL (0),
2         CHARS           ,
2         NUM_CHARS       : INITIAL (0);
2
2 ! Check for dumb calls
2 !
2 IF (.FRAME_LENGTH LEQ 0) OR (.FRAME_WIDTH LEQ 0) OR
2    (.FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]) OR
3    (.FRAME_WIDTH GTR .SCB[PSM$L_PAGE_WIDTH])
2 THEN
2     RETURN;
2
2 LOC_FRAME_LENGTH = .FRAME_LENGTH;
2
2 LOC_FRAME_WIDTH  = .FRAME_WIDTH;
2
2 PTR = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]];
2 TEMP_PTR = .CHAR_STRING[ADDR];
2 CHARS = CH$PTR(.TEMP_PTR);
```

```
; 5184          6089  2
; 5185          6090  2          INCR L FROM 0 TO (.LOC_FRAME_LENGTH-1)  DO
; 5186          6091  3              BEGIN
; 5187          6092  3              PTR = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]]        ! Address calc. is based
; 5188          6093  3                    + (.L * .SCB[PSM$L_PAGE_WIDTH]);             !  on Form Width
; 5189          6094  3              !
; 5190          6095  3              !  Move the rest of the string into the beginning of the next frame
; 5191          6096  3              !
; 5192          6097  4              IF (.NUM_CHARS LEQ .CHAR_STRING[SIZE])
; 5193          6098  4                  AND ((.CHAR_STRING[SIZE]-.NUM_CHARS) LSS .LOC_FRAME_WIDTH)
; 5194          6099  3              THEN                                                   ! Scroll it
; 5195          6100  4                  BEGIN
; 5196          6101  4                  TEMP_PTR = .CHAR_STRING[ADDR] + .NUM_CHARS; ! move remainder of str.
; 5197          6102  4                  CH$MOVE(.CHAR_STRING[SIZE]- .NUM_CHARS, .TEMP_PTR, .PTR);
; 5198          6103  4                  PTR = .PTR + (.CHAR_STRING[SIZE] - .NUM_CHARS);
; 5199          6104  4                                                                   ! incr by no. inserted
; 5200          6105  4                  START_CNT = .CHAR_STRING[SIZE] - .NUM_CHARS;
; 5201          6106  4                  TEMP_PTR = .CHAR_STRING[ADDR];
; 5202          6107  3                  END;
; 5203          6108  3
; 5204          6109  3              INCR I FROM .START_CNT TO .LOC_FRAME_WIDTH BY .CHAR_STRING[SIZE] DO
; 5205          6110  4                  BEGIN
; 5206          6111  5                  IF .CHAR_STRING[SIZE] GEQ (.LOC_FRAME_WIDTH - .I)
; 5207          6112  4                  THEN
; 5208          6113  5                      NUM_CHARS = (.LOC_FRAME_WIDTH - .I)
; 5209          6114  4                  ELSE
; 5210          6115  4                      NUM_CHARS = .CHAR_STRING[SIZE];
; 5211          6116  4
; 5212          6117  4                  CH$MOVE(.NUM_CHARS, .TEMP_PTR, .PTR);
; 5213          6118  4
; 5214          6119  4                  PTR = .PTR + .NUM_CHARS;
; 5215          6120  3                  END;
; 5216          6121  2              END;
; 5217          6122  1 END;



                                            0FFC 00000 SCROLL_FRAME:
                                                                  .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11                    ; 6057
                        5E               0C  C2 00002             SUBL2    #12, SP                                                  :
                                         59  D4 00005             CLRL     START_CNT                                                ; 6064
                                         7E  D4 00007             CLRL     NUM_CHARS                                                :
                        51       14      AC  D0 00009             MOVL     FRAME_LENGTH, R1                                         ; 6076
                                         03  15 0000D             BLEQ     1$                                                       :
                                 10      AC  D5 0000F             TSTL     FRAME_WIDTH                                              :
                                         01  14 00012 1$:         BGTR     2$                                                       :
                                             04 00014             RET                                                               :
                        50       04      AC  D0 00015 2$:         MOVL     SCB, R0                                                  ; 6077
                 01F8   C0               51  D1 00019             CMPL     R1, 504(R0)                                              :
                                         0A  14 0001E             BGTR     3$                                                       :
                        50       04      AC  D0 00020             MOVL     SCB, R0                                                  ; 6078
                 0200   C0       10      AC  D1 00024             CMPL     FRAME_WIDTH, 512(R0)                                     :
                                         01  15 0002A 3$:         BLEQ     4$                                                       :
                                             04 0002C             RET                                                               :
                   0C   AE               51  D0 0002D 4$:         MOVL     R1, LOC_FRAME_LENGTH                                     ; 6082
```

```
                              58        10  AC  D0 00031        MOVL    FRAME_WIDTH, LOC_FRAME_WIDTH              ; 6084
                        04    AE        0C  AC  D0 00035        MOVL    FRAME_PTR, PTR                            ; 6086
                              5A        08  AC  D0 0003A        MOVL    CHAR_STRING, R10                          ; 6087
                              5B        04  AA  D0 0003E        MOVL    4(R10), TEMP_PTR                          ;
                              50            5B  D0 00042        MOVL    TEMP_PTR, CHARS                           ; 6088
                        08    AE        04  AC  D0 00045        MOVL    SCB, 8(SP)                                ; 6093
                              57            01  CE 0004A        MNEGL   #1, L                                     ; 6109
                                            61  11 0004D        BRB     11$                                       ;
                  51    08    AE 00000200   8F  C1 0004F 5$:    ADDL3   #512, 8(SP), R1                           ; 6093
                  50          57            61  C5 00058        MULL3   (R1), L, R0                               ;
                        04    AE        0C BC40  9E 0005C        MOVAB   @FRAME_PTR[R0], PTR                       ;
                              6A            6E  D1 00062        CMPL    NUM_CHARS, (R10)                          ; 6097
                                            22  14 00065        BGTR    6$                                        ;
                  56          6A            6E  C3 00067        SUBL3   NUM_CHARS, (R10), R6                      ; 6098
                              58            56  D1 0006B        CMPL    R6, LOC_FRAME_WIDTH                       ;
                                            19  18 0006E        BGEQ    6$                                        ;
                  5B    04    AA            6E  C1 00070        ADDL3   NUM_CHARS, 4(R10), TEMP_PTR               ; 6101
                  56          6A            6E  C3 00075        SUBL3   NUM_CHARS, (R10), R6                      ; 6102
            04    BE          6B            56  28 00079        MOVC3   R6, (TEMP_PTR), @PTR                      ;
                        04    AE            56  C0 0007E        ADDL2   R6, PTR                                   ; 6103
                              59            56  D0 00082        MOVL    R6, START_CNT                             ; 6105
                              5B        04  AA  D0 00085        MOVL    4(R10), TEMP_PTR                          ; 6106
                              56            59  D0 00089 6$:    MOVL    START_CNT, I                              ; 6117
                                            1D  11 0008C        BRB     10$                                       ;
                  50          58            56  C3 0008E 7$:    SUBL3   I, LOC_FRAME_WIDTH, R0                    ; 6111
                              50            6A  D1 00092        CMPL    (R10), R0                                 ;
                                            05  19 00095        BLSS    8$                                        ;
                              6E            50  D0 00097        MOVL    R0, NUM_CHARS                             ; 6113
                                            03  11 0009A        BRB     9$                                        ;
                              6E            6A  D0 0009C 8$:    MOVL    (R10), NUM_CHARS                          ; 6115
            04    BE          6B            6E  28 0009F 9$:    MOVC3   NUM_CHARS, (TEMP_PTR), @PTR               ; 6117
                        04    AE            6E  C0 000A4        ADDL2   NUM_CHARS, PTR                            ; 6119
                              56            6A  C0 000A8        ADDL2   (R10), I                                  ; 6109
                              58            56  D1 000AB 10$:   CMPL    I, LOC_FRAME_WIDTH                        ;
                                            DE  15 000AE        BLEQ    7$                                        ;
                  9A          57        0C  AE  F2 000B0 11$:   AOBLSS  LOC_FRAME_LENGTH, L, 5$                   ; 6090
                                                04 000B5        RET                                               ; 6122

; Routine Size:  182 bytes,    Routine Base:  CODE + 2CF7
```

```
1 %sbttl 'MOVE_FRAME - Move Information into this Frame of the Page'
1 !++
1 ! Functional Description:
1 !       This procedure inserts a string(frame) into an array(frame).
1 !       Insertion continues until either no more string or no more room.
1 !
1 ! Formal Parameters:
1 !       SCB             - Address of the SCB
1 !       CHAR_STRING     - Descriptor of String to Insert
1 !       FRAME_PTR       - Address of first byte of Frame
1 !       FRAME_LENGTH    - Length of Frame
1 !       FRAME_WIDTH     - Width of Frame
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               Truncation is possible.
1 !
1 !--
1 ROUTINE MOVE_FRAME (
1         SCB             : REF $BBLOCK,
1         CHAR_STRING     : REF VECTOR[2],
1         FRAME_PTR       : REF PAGE_ARRAY,
1         FRAME_WIDTH     ,                 ! Number of Columns
1         FRAME_LENGTH                      ! Number of Rows
1         )  :    NOVALUE =
2 BEGIN
2
2 LOCAL   PTR             : REF PAGE_ARRAY,
2         LOC_FRAME_LENGTH,
2         LOC_FRAME_WIDTH ,
2         STR_PTR         ,
2         CURR_SIZE       ,
2         NUM_CHARS;
2
2 ! Check for dumb calls
2 !
3 IF (.FRAME_LENGTH LEQ 0) OR (.FRAME_WIDTH LEQ 0)
2 THEN
2     RETURN;
2 IF  (.FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]) OR
3     (.FRAME_WIDTH GTR .SCB[PSM$L_PAGE_QIDTH])
2 THEN
2     RETURN;
2
2
2 ! Check page boundary conditions
2 !
2 LOC_FRAME_LENGTH = .FRAME_LENGTH;
2 IF .LOC_FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]
```

```
; 5276      6180  2 THEN
; 5277      6181  2     LOC_FRAME_LENGTH = .SCB[PSM$L_PAGE_LENGTH];              ! stay in page bounds
; 5278      6182  2
; 5279      6183  2 LOC_FRAME_WIDTH  = .FRAME_WIDTH;
; 5280      6184  2 IF .LOC_FRAME_WIDTH GTR .SCB[PSM$L_PAGE_WIDTH]
; 5281      6185  2 THEN
; 5282      6186  2     LOC_FRAME_WIDTH = .SCB[PSM$L_PAGE_WIDTH];                ! stay in page bounds
; 5283      6187  2
; 5284      6188  2 !  Get string info
; 5285      6189  2 CURR_SIZE       = .CHAR_STRING[SIZE];
; 5286      6190  2 PTR             = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]];
; 5287      6191  2 STR_PTR         = .CHAR_STRING[ADDR];
; 5288      6192  2
; 5289      6193  2 !  Do a quick fill of the frame
; 5290      6194  2  FILL_FRAME (.SCB,
; 5291      6195  2               %CHAR(32),         ! fill with blanks
; 5292      6196  2               FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
; 5293      6197  2               .LOC_FRAME_WIDTH,
; 5294      6198  2               .LOC_FRAME_LENGTH);
; 5295      6199  2
; 5296      6200  2 DECRU L FROM .LOC_FRAME_LENGTH TO 1 DO
; 5297      6201  3     BEGIN
; 5298      6202  3     IF .CURR_SIZE GEQ .LOC_FRAME_WIDTH THEN
; 5299      6203  3         NUM_CHARS = .LOC_FRAME_WIDTH
; 5300      6204  3     ELSE
; 5301      6205  3         NUM_CHARS = .CURR_SIZE;
; 5302      6206  3
; 5303      6207  3     CH$MOVE(.NUM_CHARS, .STR_PTR, .PTR);
; 5304      6208  3
; 5305      6209  3     PTR = .PTR + .SCB[PSM$L_PAGE_WIDTH];                    ! Address calc. is based
; 5306      6210  3                                                             !  on Form Width
; 5307      6211  3     STR_PTR = .STR_PTR + .LOC_FRAME_WIDTH;
; 5308      6212  3
; 5309      6213  3     IF .LOC_FRAME_WIDTH GTRU .CURR_SIZE THEN
; 5310      6214  3         EXITLOOP;
; 5311      6215  3
; 5312      6216  3     CURR_SIZE = .CURR_SIZE - .NUM_CHARS;                    ! Decrease string size
; 5313      6217  2     END;
; 5314      6218  1 END;



                                    0FFC 00000 MOVE_FRAME:
                                                     .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11          ; 6149
                 5E           04  C2 00002           SUBL2   #4, SP                                        :
                 51     14    AC  D0 00005           MOVL    FRAME_LENGTH, R1                              ; 6167
                              7C  15 00009           BLEQ    7$                                            :
                        10    AC  D5 0000B           TSTL    FRAME_WIDTH                                   :
                              77  15 0000E           BLEQ    7$                                            :
                 57     04    AC  D0 00010           MOVL    SCB, R7                                       ; 6170
                 50   01F8    C7  D0 00014           MOVL    504(R7), R0                                   :
                 50           51  D1 00019           CMPL    R1, R0                                        :
                              69  14 0001C           BGTR    7$                                            :
          0200   C7     10    AC  D1 0001E           CMPL    FRAME_WIDTH, 512(R7)                          ; 6171
                              61  14 00024           BGTR    7$                                            :
```

```
                                        52             51 D0 00026          MOVL    R1, LOC_FRAME_LENGTH                          ; 6178
                                        50             52 D1 00029          CMPL    LOC_FRAME_LENGTH, R0                          ; 6179
                                                       03 15 0002C          BLEQ    1$                                            ;
                                        52             50 D0 0002E          MOVL    R0, LOC_FRAME_LENGTH                          ; 6181
                                        58        10   AC D0 00031 1$:      MOVL    FRAME_WIDTH, LOC_FRAME_WIDTH                  ; 6183
                                 0200   C7             58 D1 00035          CMPL    LOC_FRAME_WIDTH, 512(R7)                      ; 6184
                                                       05 15 0003A          BLEQ    2$                                            ;
                                        58      0200   C7 D0 0003C          MOVL    512(R7), LOC_FRAME_WIDTH                      ; 6186
                                        50        08   AC D0 00041 2$:      MOVL    CHAR_STRING, R0                               ; 6189
                                        6E        0C   AC D0 00045          MOVL    FRAME_PTR, PTR                                ; 6190
                                        5A             60 7D 00049          MOVQ    (R0), CURR_SIZE                               ; 6189
                                                       52 DD 0004C          PUSHL   LOC_FRAME_LENGTH                              ; 6198
                                                       58 DD 0004E          PUSHL   LOC_FRAME_WIDTH                               ; 6197
                                                  0C   AC DD 00050          PUSHL   FRAME_PTR                                     ; 6196
                                                       20 DD 00053          PUSHL   #32                                           ;
                                                       57 DD 00055          PUSHL   R7                                            ;
                                 FE99   CF             05 FB 00057          CALLS   #5, FILL_FRAME                                ;
                                        56             52 D0 0005C          MOVL    LOC_FRAME_LENGTH, L                           ; 6200
                                                       24 11 0005F          BRB     6$                                            ;
                                        58             5A D1 00061 3$:      CMPL    CURR_SIZE, LOC_FRAME_WIDTH                    ; 6202
                                                       05 19 00064          BLSS    4$                                            ;
                                        59             58 D0 00066          MOVL    LOC_FRAME_WIDTH, NUM_CHARS                    ; 6203
                                                       03 11 00069          BRB     5$                                            ;
                                        59             5A D0 0006B 4$:      MOVL    CURR_SIZE, NUM_CHARS                          ; 6205
                     00   BE            6B             59 28 0006E 5$:      MOVC3   NUM_CHARS, (STR_PTR), @PTR                    ; 6207
                                        6E      0200   C7 C0 00073          ADDL2   512(R7), PTR                                  ; 6209
                                        5B             58 C0 00078          ADDL2   LOC_FRAME_WIDTH, STR_PTR                      ; 6211
                                        5A             58 D1 0007B          CMPL    LOC_FRAME_WIDTH, CURR_SIZE                    ; 6213
                                                       07 1A 0007E          BGTRU   7$                                            ;
                                        5A             59 C2 00080          SUBL2   NUM_CHARS, CURR_SIZE                          ; 6216
                                                       56 D7 00083          DECL    L                                             ; 6200
                                                       DA 12 00085 6$:      BNEQ    3$                                            ;
                                                          04 00087 7$:      RET                                                   ; 6218

; Routine Size:  136 bytes,    Routine Base:  CODE + 2DAD
; 5316      6219  1 %sbttl 'INSERT_FRAME - Insert Information into this Frame of the Page'
; 5317      6220  1 !++
; 5318      6221  1 ! Functional Description:
; 5319      6222  1 !       This procedure inserts a string into an array(frame).
; 5320      6223  1 !       Insertion continues until either no more string or no more room
; 5321      6224  1 !       Delimiting characters are used to correctly parse the string
; 5322      6225  1 !       prior to insertion.
; 5323      6226  1 !
; 5324      6227  1 ! Formal Parameters:
; 5325      6228  1 !       SCB             - Address of the SCB
; 5326      6229  1 !       CHAR_STRING     - Descriptor of String to Insert
; 5327      6230  1 !       FRAME_PTR       - Address of first byte of Frame
; 5328      6231  1 !       FRAME_LENGTH    - Length of Frame
; 5329      6232  1 !       FRAME_WIDTH     - Width of Frame
; 5330      6233  1 !
; 5331      6234  1 ! Implicit Inputs:
; 5332      6235  1 !               none
; 5333      6236  1 !
; 5334      6237  1 ! Implicit Outputs:
; 5335      6238  1 !               none
; 5336      6239  1 !
; 5337      6240  1 ! Returned Value:
; 5338      6241  1 !               none
; 5339      6242  1 !
; 5340      6243  1 ! Side Effects:
; 5341      6244  1 !               Truncation is possible.
; 5342      6245  1 !
; 5343      6246  1 !--
; 5344      6247  1 ROUTINE INSERT_FRAME (
; 5345      6248  1         SCB             : REF $BBLOCK,
; 5346      6249  1         CHAR_STRING     : REF VECTOR[2],
; 5347      6250  1         FRAME_PTR       : REF PAGE_ARRAY,
; 5348      6251  1         FRAME_WIDTH     ,               ! Number of Columns
; 5349      6252  1         FRAME_LENGTH                    ! Number of Rows
; 5350      6253  1         )  :    NOVALUE =
; 5351      6254  2 BEGIN
; 5352      6255  2 BUILTIN AP;                     ! just in case trucation occurred ... don't delimit
; 5353      6256  2
; 5354      6257  2 LITERAL
; 5355      6258  2         LEADING         = 0,
; 5356      6259  2         TRAILING        = 1;
; 5357      6260  2
; 5358      6261  2 LOCAL   PTR             : REF PAGE_ARRAY,
; 5359      6262  2         LOC_FRAME_LENGTH,
; 5360      6263  2         LOC_FRAME_WIDTH ,
; 5361      6264  2         STR_PTR         ,
; 5362      6265  2         CURR_SIZE       ,
; 5363      6266  2         DUM_LEN         ,
; 5364      6267  2         NUM_CHARS;
; 5365      6268  2
; 5366      6269  2 ! Check for dumb calls
; 5367      6270  2 !
; 5368      6271  3 IF (.FRAME_LENGTH LEQ 0) OR (.FRAME_WIDTH LEQ 0)
; 5369      6272  2 THEN
; 5370      6273  2     RETURN;
; 5371      6274  2 IF  (.FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]) OR
; 5372      6275  3     (.FRAME_WIDTH GTR .SCB[PSM$L_PAGE_WIDTH])
```

```
2 THEN
2     RETURN;
2
2 ! Check page boundary conditions
2 !
2 LOC_FRAME_LENGTH = .FRAME_LENGTH;
2 IF .LOC_FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]
2 THEN
2     LOC_FRAME_LENGTH = .SCB[PSM$L_PAGE_LENGTH];                ! stay in page bounds
2
2 LOC_FRAME_WIDTH  = .FRAME_WIDTH;
2 IF .LOC_FRAME_WIDTH GTR .SCB[PSM$L_PAGE_WIDTH]
2 THEN
2     LOC_FRAME_WIDTH = .SCB[PSM$L_PAGE_WIDTH];                  ! stay in page bounds
2
2 !  Get string info
2 CURR_SIZE          = .CHAR_STRING[SIZE];
2 PTR                = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]];
2 STR_PTR            = .CHAR_STRING[ADDR];
2
2 !  Do a quick fill of the frame
2  FILL_FRAME (.SCB,
2              %CHAR(32),          ! fill with blanks
2              FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
2              .LOC_FRAME_WIDTH,
2              .LOC_FRAME_LENGTH);
2
2 DECR L FROM .LOC_FRAME_LENGTH TO 1 DO
3     BEGIN
3     IF .CURR_SIZE GEQ .LOC_FRAME_WIDTH THEN
3         NUM_CHARS = DELIMIT_STRING(.STR_PTR,%CHAR(32),.LOC_FRAME_WIDTH)
3     ELSE
3         NUM_CHARS = .CURR_SIZE;
3                                     ! adjust pointer
3     DISCARD (LEADING, %C' ', .STR_PTR, .NUM_CHARS, NUM_CHARS, STR_PTR);
3     CH$MOVE(.NUM_CHARS, .STR_PTR, .PTR);
3
3     PTR = .PTR + .SCB[PSM$L_PAGE_WIDTH];                        ! Address calc. is based
3                                                                 !  on Form Width
3     STR_PTR = .STR_PTR + .NUM_CHARS;
3
3     IF .LOC_FRAME_WIDTH GEQ .CURR_SIZE THEN
3         EXITLOOP;
3
3     CURR_SIZE = .CURR_SIZE - .NUM_CHARS;                        ! Decrease string size
2     END;
2
3 IF .STR_PTR LSS (.CHAR_STRING[ADDR] + .CHAR_STRING[SIZE])
2 THEN          ! truncation occurred
2     CALLG (.AP, MOVE_FRAME);                                   ! dont delimit..just move str.
1 END;



                                    0FFC 00000 INSERT_FRAME:
```

```
                                               .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11        ; 6247
              5E            08 C2 00002        SUBL2   #8, SP                                      ;
              51      14    AC D0 00005        MOVL    FRAME_LENGTH, R1                            ; 6271
                            03 15 00009        BLEQ    1$                                          ;
                      10    AC D5 0000B        TSTL    FRAME_WIDTH                                 ;
                            01 14 0000E 1$:    BGTR    2$                                          ;
                               04 00010        RET                                                 ;
              58      04    AC D0 00011 2$:    MOVL    SCB, R8                                     ; 6274
              50    01F8    C8 D0 00015        MOVL    504(R8), R0                                 ;
              50            51 D1 0001A        CMPL    R1, R0                                      ;
                            06 14 0001D        BGTR    3$                                          ;
       0200   C8      10    AC D1 0001F        CMPL    FRAME_WIDTH, 512(R8)                        ; 6275
                            01 15 00025 3$:    BLEQ    4$                                          ;
                               04 00027        RET                                                 ;
              52            51 D0 00028 4$:    MOVL    R1, LOC_FRAME_LENGTH                        ; 6281
              50            52 D1 0002B        CMPL    LOC_FRAME_LENGTH, R0                        ; 6282
                            03 15 0002E        BLEQ    5$                                          ;
              52            50 D0 00030        MOVL    R0, LOC_FRAME_LENGTH                        ; 6284
              59      10    AC D0 00033 5$:    MOVL    FRAME_WIDTH, LOC_FRAME_WIDTH                ; 6286
       0200   C8            59 D1 00037        CMPL    LOC_FRAME_WIDTH, 512(R8)                    ; 6287
                            05 15 0003C        BLEQ    6$                                          ;
              59    0200    C8 D0 0003E        MOVL    512(R8), LOC_FRAME_WIDTH                    ; 6289
              57      08    AC D0 00043 6$:    MOVL    CHAR_STRING, R7                             ; 6292
              5A            67 D0 00047        MOVL    (R7), CURR_SIZE                             ;
              5B      0C    AC D0 0004A        MOVL    FRAME_PTR, PTR                              ; 6293
              6E      04    A7 D0 0004E        MOVL    4(R7), STR_PTR                              ; 6294
                            52 DD 00052        PUSHL   LOC_FRAME_LENGTH                            ; 6301
                            59 DD 00054        PUSHL   LOC_FRAME_WIDTH                             ; 6300
                      0C    AC DD 00056        PUSHL   FRAME_PTR                                   ; 6299
                            20 DD 00059        PUSHL   #32                                         ;
                            58 DD 0005B        PUSHL   R8                                          ;
       FE0B   CF            05 FB 0005D        CALLS   #5, FILL_FRAME                              ;
              56      01    A2 9E 00062        MOVAB   1(R2), L                                    ; 6303
                            47 11 00066        BRB     10$                                         ;
              59            5A D1 00068 7$:    CMPL    CURR_SIZE, LOC_FRAME_WIDTH                  ; 6305
                            12 19 0006B        BLSS    8$                                          ;
                            59 DD 0006D        PUSHL   LOC_FRAME_WIDTH                             ; 6306
                            20 DD 0006F        PUSHL   #32                                         ;
                      08    AE DD 00071        PUSHL   STR_PTR                                     ;
      0000V   CF            03 FB 00074        CALLS   #3, DELIMIT_STRING                          ;
         04   AE            50 D0 00079        MOVL    R0, NUM_CHARS                               ;
                            04 11 0007D        BRB     9$                                          ;
         04   AE            5A D0 0007F 8$:    MOVL    CURR_SIZE, NUM_CHARS                        ; 6308
                            5E DD 00083 9$:    PUSHL   SP                                          ; 6310
                      08    AE 9F 00085        PUSHAB  NUM_CHARS                                   ;
                      0C    AE DD 00088        PUSHL   NUM_CHARS                                   ;
                      0C    AE DD 0008B        PUSHL   STR_PTR                                     ;
                            20 DD 0008E        PUSHL   #32                                         ;
                            7E D4 00090        CLRL    -(SP)                                       ;
      0000V   CF            06 FB 00092        CALLS   #6, DISCARD                                 ;
6B       00   BE      04    AE 28 00097        MOVC3   NUM_CHARS, @STR_PTR, (PTR)                  ; 6311
              5B    0200    C8 C0 0009D        ADDL2   512(R8), PTR                                ; 6313
              6E      04    AE C0 000A2        ADDL2   NUM_CHARS, STR_PTR                          ; 6315
              5A            59 D1 000A6        CMPL    LOC_FRAME_WIDTH, CURR_SIZE                  ; 6317
                            07 18 000A9        BGEQ    11$                                         ;
              5A      04    AE C2 000AB        SUBL2   NUM_CHARS, CURR_SIZE                        ; 6320
              B6            56 F5 000AF 10$:   SOBGTR  L, 7$                                       ; 6303
```

```
                    57      04    A7             67  C1 000B2 11$:      ADDL3   (R7), 4(R7), R7                                     ; 6323
                                  57             6E  D1 000B7           CMPL    STR_PTR, R7                                         ;
                                                 05  18 000BA           BGEQ    12$                                                 ;
                          FEB7    CF             6C  FA 000BC           CALLG   (AP), MOVE_FRAME                                    ; 6325
                                                     04 000C1 12$:      RET                                                         ; 6326

; Routine Size:  194 bytes,     Routine Base:  CODE + 2E35
```

```
1 %sbttl 'CENTER_FRAME - Insert String Information into the Center of this Frame'
1 !++
1 ! Functional Description:
1 !         This procedure inserts a string into the cneter of an array(frame).
1 !         Insertion continues until either no more string or no more room
1 !         The idea is to center the string within the frame.
1 !         1) If the string is shorter than the entire frame then
1 !            center the string in the frame.
1 !         2) If the string is longer than the entire frame then
1 !            insert as much of the string as possible.
1 !         3) Use an assumed pad of blanks for beginning and end of string.
1 !
1 !
1 ! Formal Parameters:
1 !         SCB             - Address of the SCB
1 !         CHAR_STRING     - Descriptor of String to Insert
1 !         FRAME_PTR       - Address of first byte of Frame
1 !         FRAME_LENGTH    - Length of Frame
1 !         FRAME_WIDTH     - Width of Frame
1 !
1 ! Implicit Inputs:
1 !                   none
1 !
1 ! Implicit Outputs:
1 !                   none
1 !
1 ! Returned Value:
1 !                   none
1 !
1 ! Side Effects:
1 !                   Truncation is possible.
1 !
1 !--
1 ROUTINE CENTER_FRAME (
1         SCB             : REF $BBLOCK,
1         CHAR_STRING     : REF VECTOR[2],
1         FRAME_PTR       : REF PAGE_ARRAY,
1         FRAME_WIDTH     ,                   ! Number of Columns
1         FRAME_LENGTH                        ! Number of Rows
1         )  :    NOVALUE =
2 BEGIN
2 LITERAL K_MAX_BUFFER_SIZE = 512, ! maximum possible buffer size
2         K_PAD_LEN         =   2; ! length to pad the string
2
2 LOCAL   PTR             : REF PAGE_ARRAY, ! points to col 0, variable row
2         LOC_FRAME_LENGTH,                 ! local count of frame_length
2         LOC_FRAME_WIDTH ,                 ! local count of frame_width
2         PAD_CHAR        ,                 ! pad character is assumed a space
2         PTR_OFFSET      : SIGNED,         ! pos/neg offset to origin
2         STR_PTR         ,                 ! points to str position in frame
2         STR_SIZE        ,                 ! num char left in string
2         BUFFER          : VECTOR [512,byte],  ! variable buff for pad and string
2         STR_DESC        : VECTOR [2];     ! desc of string
2
2 ! Check for dumb calls. Frame dimensions must be writable
2 !
3 IF (.FRAME_LENGTH LEQ 0) OR (.FRAME_WIDTH LEQ 0)
```

```
; 5482          6384   2 THEN
; 5483          6385   2     RETURN;
; 5484          6386   2 IF  (.FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]) OR
; 5485          6387   3     (.FRAME_WIDTH GTR .SCB[PSM$L_PAGE_WIDTH])
; 5486          6388   2 THEN
; 5487          6389   2     RETURN;
; 5488          6390   2 ! String must not be zero !!
; 5489          6391   2 !
; 5490          6392   2 IF .CHAR_STRING[SIZE] EQL 0 THEN RETURN SS$_NORMAL;
; 5491          6393   2
; 5492          6394   2 ! Check page boundary conditions
; 5493          6395   2 !
; 5494          6396   2 LOC_FRAME_LENGTH = .FRAME_LENGTH;
; 5495          6397   2 IF .LOC_FRAME_LENGTH GTR .SCB[PSM$L_PAGE_LENGTH]
; 5496          6398   2 THEN
; 5497          6399   2     LOC_FRAME_LENGTH = .SCB[PSM$L_PAGE_LENGTH];          ! stay in page bounds
; 5498          6400   2
; 5499          6401   2 LOC_FRAME_WIDTH  = .FRAME_WIDTH;
; 5500          6402   2 IF .LOC_FRAME_WIDTH GTR .SCB[PSM$L_PAGE_WIDTH]
; 5501          6403   2 THEN
; 5502          6404   2     LOC_FRAME_WIDTH = .SCB[PSM$L_PAGE_WIDTH];            ! stay in page bounds
; 5503          6405   2
; 5504          6406   2 !  Get string into padding buffer if enough room
; 5505          6407   2 !
; 5506          6408   2 STR_DESC[SIZE] = %ALLOCATION (BUFFER);  ! length of string and pad chars
; 5507          6409   2 STR_DESC[ADDR] = BUFFER;                ! ptr into pad&string buffer
; 5508          6410   2 PAD_CHAR  = %CHAR(32,32);
; 5509          6411   2 PTR = FRAME_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]];    ! init
; 5510          6412   2
; 5511          6413   2 ! Pad the string if there is enough room
; 5512          6414   3 IF (.CHAR_STRING[SIZE] + (2 * K_PAD_LEN))
; 5513          6415   2                 LEQ
; 5514          6416   3         (.LOC_FRAME_WIDTH * .LOC_FRAME_LENGTH)
; 5515          6417   2 THEN
; 5516          6418   3     BEGIN
; 5517          6419   3     !  Set the size to correct value
; 5518          6420   3     !
; 5519          6421   3     IF .CHAR_STRING[SIZE] LEQ (K_MAX_BUFFER_SIZE-2*(K_PAD_LEN)) THEN
; 5520          6422   4         STR_DESC[SIZE] = .CHAR_STRING[SIZE] + ( 2 * (K_PAD_LEN))
; 5521          6423   3     ELSE
; 5522          6424   3         STR_DESC[SIZE] = K_MAX_BUFFER_SIZE;
; 5523          6425   3
; 5524          6426   3     !  Pad the string using the local buffer
; 5525          6427   3     CH$COPY (K_PAD_LEN, PAD_CHAR, .CHAR_STRING[SIZE], .CHAR_STRING[ADDR],
; 5526          6428   3              K_PAD_LEN, PAD_CHAR, .PAD_CHAR, .STR_DESC[SIZE],
; 5527          6429   3              .STR_DESC[ADDR]);
; 5528          6430   3     END
; 5529          6431   2 ELSE              ! copy into local buffer
; 5530          6432   3     BEGIN
; 5531          6433   3     CH$MOVE(.CHAR_STRING[SIZE], .CHAR_STRING[ADDR], .STR_DESC[ADDR]);
; 5532          6434   3     STR_DESC[SIZE] = .CHAR_STRING[SIZE];
; 5533          6435   2     END;
; 5534          6436   2
; 5535          6437   2 !  Calc offset to pointer using new padded length
; 5536          6438   3 IF ( .LOC_FRAME_WIDTH * .LOC_FRAME_LENGTH) GTR (.STR_DESC[SIZE])
; 5537          6439   2 THEN
; 5538          6440   4     PTR_OFFSET  = (( .LOC_FRAME_WIDTH * .LOC_FRAME_LENGTH)
; 5539          6441  2                                   - (.STR_DESC[SIZE]))/2
; 5540          6442  2 ELSE
; 5541          6443  2     PTR_OFFSET  = 0;
; 5542          6444  2
; 5543          6445  2 !  Check for negative offset
; 5544          6446  2 IF .PTR_OFFSET LSS 0
; 5545          6447  2 THEN
; 5546          6448  2     PTR_OFFSET = 0;
; 5547          6449  2
; 5548          6450  2 !  Set pointer to buffer
; 5549          6451  2 STR_PTR = .PTR + .PTR_OFFSET;
; 5550          6452  2 STR_SIZE = .STR_DESC[SIZE];
; 5551          6453  2
; 5552          6454  2 DECRU L FROM .LOC_FRAME_LENGTH TO 1 DO
; 5553          6455  3     BEGIN
; 5554          6456  3
; 5555          6457  3     LOCAL NUM_CHARS; ! number of chars to move to the frame
; 5556          6458  3
; 5557          6459  3     IF .PTR_OFFSET GEQ .LOC_FRAME_WIDTH THEN
; 5558          6460  4         BEGIN
; 5559          6461  4         PTR = .PTR + .SCB[PSM$L_PAGE_WIDTH];        ! go to next row of frame
; 5560          6462  4         PTR_OFFSET = .PTR_OFFSET - .LOC_FRAME_WIDTH; ! adjust offset to column
; 5561          6463  4         END
; 5562          6464  4     ELSE BEGIN
; 5563          6465  4
; 5564          6466  4         IF .STR_SIZE GEQ (.LOC_FRAME_WIDTH - .PTR_OFFSET) THEN
; 5565          6467  4             NUM_CHARS = .LOC_FRAME_WIDTH - .PTR_OFFSET
; 5566          6468  4         ELSE                            ! check for overflow of frame width
; 5567          6469  4             NUM_CHARS = .STR_SIZE;              ! - insert which ever is less
; 5568          6470  4
; 5569          6471  4         CH$MOVE(.NUM_CHARS, .STR_DESC[ADDR], .STR_PTR);
; 5570          6472  4         STR_PTR = .PTR + .SCB[PSM$L_PAGE_WIDTH];! Address calc. is base on
; 5571          6473  4         PTR = .STR_PTR;                     !    Frame ptr[0,0] & Form Width
; 5572          6474  4
; 5573          6475  4         STR_DESC[ADDR] = .STR_DESC[ADDR] + .NUM_CHARS;
; 5574          6476  4
; 5575          6477  4         IF .LOC_FRAME_WIDTH GTRU .STR_SIZE      ! Already inserted it all
; 5576          6478  4         THEN
; 5577          6479  4             EXITLOOP;
; 5578          6480  4
; 5579          6481  4         STR_SIZE = .STR_SIZE - .NUM_CHARS;                  ! Decrease string size
; 5580          6482  4
; 5581          6483  4      ! reset the offset to start column one
; 5582          6484  4         PTR_OFFSET = 0;
; 5583          6485  4
; 5584          6486  3         END;
; 5585          6487  2     END;
; 5586          6488  1 END;



                                                  0FFC 00000 CENTER_FRAME:
                                                             .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11           ; 6360
                          5E      FDE8   CE  9E 00002        MOVAB    -536(SP), SP                                   ;
                          52        14   AC  D0 00007        MOVL     FRAME_LENGTH, R2                               ; 6383
```

```
                                         03 15 0000B       BLEQ    1$                                           ;
                                   10    AC D5 0000D       TSTL    FRAME_WIDTH                                  ;
                                         01 14 00010 1$:   BGTR    2$                                           ;
                                         04    00012       RET                                                  ;
                           50      04    AC D0 00013 2$:   MOVL    SCB, R0                                      ; 6386
                           51    01F8    C0 D0 00017       MOVL    504(R0), R1                                  ;
                           51            52 D1 0001C       CMPL    R2, R1                                       ;
                                         06 14 0001F       BGTR    3$                                           ;
                    0200   C0      10    AC D1 00021       CMPL    FRAME_WIDTH, 512(R0)                         ; 6387
                                         01 15 00027 3$:   BLEQ    4$                                           ;
                                         04    00029       RET                                                  ;
                           5A      08    BC D0 0002A 4$:   MOVL    @CHAR_STRING, R10                            ; 6392
                                         01 12 0002E       BNEQ    5$                                           ;
                                         04    00030       RET                                                  ;
                           56            52 D0 00031 5$:   MOVL    R2, LOC_FRAME_LENGTH                         ; 6396
                           51            56 D1 00034       CMPL    LOC_FRAME_LENGTH, R1                         ; 6397
                                         03 15 00037       BLEQ    6$                                           ;
                           56            51 D0 00039       MOVL    R1, LOC_FRAME_LENGTH                         ; 6399
                           57      10    AC D0 0003C 6$:   MOVL    FRAME_WIDTH, LOC_FRAME_WIDTH                 ; 6401
                      04   AE    0200    C0 9E 00040       MOVAB   512(R0), 4(SP)                               ; 6402
                      04   BE            57 D1 00046       CMPL    LOC_FRAME_WIDTH, @4(SP)                      ;
                                         04 15 0004A       BLEQ    7$                                           ;
                           57      04    BE D0 0004C       MOVL    @4(SP), LOC_FRAME_WIDTH                      ; 6404
                      10   AE    0200    8F 3C 00050 7$:   MOVZWL  #512, STR_DESC                               ; 6408
                      14   AE      18    AE 9E 00056       MOVAB   BUFFER, STR_DESC+4                           ; 6409
                      0C   AE    2020    8F 3C 0005B       MOVZWL  #8224, PAD_CHAR                              ; 6410
                      08   AE      0C    AC D0 00061       MOVL    FRAME_PTR, PTR                               ; 6411
                           58      08    AC D0 00066       MOVL    CHAR_STRING, R8                              ; 6427
                           50      04    AA 9E 0006A       MOVAB   4(R10), R0                                   ; 6414
             6E            57            56 C5 0006E       MULL3   LOC_FRAME_LENGTH, LOC_FRAME_WIDTH, (SP)      ; 6416
                           6E            50 D1 00072       CMPL    R0, (SP)                                     ;
                                         47 14 00075       BGTR    10$                                          ;
                000001FC   8F            5A D1 00077       CMPL    R10, #508                                    ; 6421
                                         06 14 0007E       BGTR    8$                                           ;
                      10   AE            50 D0 00080       MOVL    R0, STR_DESC                                 ; 6422
                                         06 11 00084       BRB     9$                                           ;
                      10   AE    0200    8F 3C 00086 8$:   MOVZWL  #512, STR_DESC                               ; 6424
                           5B      10    AE D0 0008C 9$:   MOVL    STR_DESC, R11                                ; 6428
                           59      14    AE D0 00090       MOVL    STR_DESC+4, R9                               ; 6429
5B      0C   AE       0C   AE            02 2C 00094       MOVC5   #2, PAD_CHAR, PAD_CHAR, R11, (R9)            ;
                                         69    0009B                                                            ;
                                         2A 18 0009C       BGEQ    11$                                          ;
                           59            02 C0 0009E       ADDL2   #2, R9                                       ;
                           5B            02 C2 000A1       SUBL2   #2, R11                                      ;
5B      0C   AE       04   B8            5A 2C 000A4       MOVC5   R10, @4(R8), PAD_CHAR, R11, (R9)             ;
                                         69    000AB                                                            ;
                                         1A 18 000AC       BGEQ    11$                                          ;
                           59            5A C0 000AE       ADDL2   R10, R9                                      ;
                           5B            5A C2 000B1       SUBL2   R10, R11                                     ;
5B      0C   AE       0C   AE            02 2C 000B4       MOVC5   #2, PAD_CHAR, PAD_CHAR, R11, (R9)            ;
                                         69    000BB                                                            ;
                                         0A 11 000BC       BRB     11$                                          ; 6414
        14   BE       04   B8            5A 28 000BE 10$:  MOVC3   R10, @4(R8), @STR_DESC+4                     ; 6433
                      10   AE            5A D0 000C4       MOVL    R10, STR_DESC                                ; 6434
                      10   AE            6E D1 000C8 11$:  CMPL    (SP), STR_DESC                               ; 6438
                                         0B 15 000CC       BLEQ    12$                                          ;
             50            6E      10    AE C3 000CE       SUBL3   STR_DESC, (SP), R0                           ; 6441
```

```
            5A            50               02  C7 000D3          DIVL3   #2, R0, PTR_OFFSET                                      ;
                                           02  11 000D7          BRB     13$                                                     ; 6440
                                           5A  D4 000D9 12$:     CLRL    PTR_OFFSET                                              ; 6443
                                           02  18 000DB 13$:     BGEQ    14$                                                     ; 6446
                                           5A  D4 000DD          CLRL    PTR_OFFSET                                              ; 6448
            5B            5A       08      AE  C1 000DF 14$:     ADDL3   PTR, PTR_OFFSET, STR_PTR                                ; 6451
                          59       10      AE  D0 000E4          MOVL    STR_DESC, STR_SIZE                                      ; 6452
                                           3F  11 000E8          BRB     20$                                                     ; 6454
                          57               5A  D1 000EA 15$:     CMPL    PTR_OFFSET, LOC_FRAME_WIDTH                             ; 6459
                                           0A  19 000ED          BLSS    16$                                                     ;
                    08    AE       04      BE  C0 000EF          ADDL2   @4(SP), PTR                                             ; 6461
                          5A               57  C2 000F4          SUBL2   LOC_FRAME_WIDTH, PTR_OFFSET                             ; 6462
                                           2E  11 000F7          BRB     19$                                                     ; 6459
            50            57               5A  C3 000F9 16$:     SUBL3   PTR_OFFSET, LOC_FRAME_WIDTH, R0                         ; 6466
                          50               59  D1 000FD          CMPL    STR_SIZE, R0                                            ;
                                           05  19 00100          BLSS    17$                                                     ;
                          58               50  D0 00102          MOVL    R0, NUM_CHARS                                           ; 6467
                                           03  11 00105          BRB     18$                                                     ;
                          58               59  D0 00107 17$:     MOVL    STR_SIZE, NUM_CHARS                                     ; 6469
            6B      14    BE               58  28 0010A 18$:     MOVC3   NUM_CHARS, @STR_DESC+4, (STR_PTR)                       ; 6471
            5B      08    AE       04      BE  C1 0010F          ADDL3   @4(SP), PTR, STR_PTR                                    ; 6472
                    08    AE               5B  D0 00115          MOVL    STR_PTR, PTR                                            ; 6473
                    14    AE               58  C0 00119          ADDL2   NUM_CHARS, STR_DESC+4                                   ; 6475
                          59               57  D1 0011D          CMPL    LOC_FRAME_WIDTH, STR_SIZE                               ; 6477
                                           0B  1A 00120          BGTRU   21$                                                     ;
                          59               58  C2 00122          SUBL2   NUM_CHARS, STR_SIZE                                     ; 6481
                                           5A  D4 00125          CLRL    PTR_OFFSET                                              ; 6484
                                           56  D7 00127 19$:     DECL    L                                                       ; 6454
                                           56  D5 00129 20$:     TSTL    L                                                       ;
                                           BD  12 0012B          BNEQ    15$                                                     ;
                                               04 0012D 21$:     RET                                                             ; 6488

; Routine Size:  302 bytes,    Routine Base:  CODE + 2EF7
; 5588          6489  1 %sbttl 'MERGE_FRAME - Merge Information in this Frame of the Page'
; 5589          6490  1 !++
; 5590          6491  1 ! Functional Description:
; 5591          6492  1 !       This procedure merges rows of non-blank strings into an array(frame).
; 5592          6493  1 !       Merging continues until either no more strings or no more frame.  The
; 5593          6494  1 !       contents of the frame are merged to the base of the frame.  Any row
; 5594          6495  1 !       with data present is considered impure and is merged.
; 5595          6496  1 !
; 5596          6497  1 ! Formal Parameters:
; 5597          6498  1 !       SCB             - Address of the SCB
; 5598          6499  1 !       CHAR_STRING     - Descriptor of String to Insert
; 5599          6500  1 !       FRAME_PTR       - Address of first byte of Frame
; 5600          6501  1 !       FRAME_LENGTH    - Length of Frame
; 5601          6502  1 !       FRAME_WIDTH     - Width of Frame
; 5602          6503  1 !
; 5603          6504  1 ! Implicit Inputs:
; 5604          6505  1 !               none
; 5605          6506  1 !
; 5606          6507  1 ! Implicit Outputs:
; 5607          6508  1 !               none
; 5608          6509  1 !
; 5609          6510  1 ! Returned Value:
; 5610          6511  1 !               none
; 5611          6512  1 !
; 5612          6513  1 ! Side Effects:
; 5613          6514  1 !               Truncation is possible.
; 5614          6515  1 !
; 5615          6516  1 !--
; 5616          6517  1 ROUTINE MERGE_FRAME (
; 5617          6518  1         SCB             : REF $BBLOCK,
; 5618          6519  1         CHAR_STRING     : REF VECTOR[2],
; 5619          6520  1         FRAME_PTR       : REF PAGE_ARRAY,
; 5620          6521  1         FRAME_WIDTH     ,               ! Number of Columns
; 5621          6522  1         FRAME_LENGTH    ,               ! Number of Rows
; 5622          6523  1         RET_LEN         : REF VECTOR[,word]
; 5623          6524  1         )  :    NOVALUE =
; 5624          6525  2 BEGIN
; 5625          6526  2
; 5626          6527  2 LITERAL K_MAX_SIZE      = 256;
; 5627          6528  2
; 5628          6529  2 LOCAL
; 5629          6530  2         CLR_STR         : VECTOR[2],
; 5630          6531  2         SRCE_STR        : VECTOR[2],
; 5631          6532  2         BUFFER          : VECTOR[256,byte],
; 5632          6533  2         CURRENT_PTR     : REF PAGE_ARRAY,
; 5633          6534  2         CURRENT_LEN     ,
; 5634          6535  2         DEST_OFFSET     ,
; 5635          6536  2         SOURCE_OFFSET   ;
; 5636          6537  2
; 5637          6538  2 ! setup clr_str
; 5638          6539  2 !
; 5639          6540  2 CLR_STR[ADDR] = .CHAR_STRING[ADDR];
; 5640          6541  2 CLR_STR[SIZE] = .FRAME_WIDTH;
; 5641          6542  2 CH$FILL ( %C' ', .FRAME_WIDTH, .CLR_STR[ADDR]);
; 5642          6543  2 ! setup source_str
; 5643          6544  2 !
; 5644          6545  2 SRCE_STR[SIZE] = %ALLOCATION(BUFFER);
```

```
2 SRCE_STR[ADDR] = BUFFER;
2
2 ! setup current point and offsets into frame
2 !
2 CURRENT_PTR = FRAME_PTR[0,0, .SCB[PSM$L_PAGE_WIDTH]];
2
2 SOURCE_OFFSET = .FRAME_LENGTH-1;                       ! start at frame boundaries
2 DEST_OFFSET = .FRAME_LENGTH-1;
2
2 DECRU I FROM (.FRAME_LENGTH-1) TO 0
2 DO
3     BEGIN
3     FIND_DEST_PTR (      .SCB,
3                          CLR_STR[0],
3                          CURRENT_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
3                          .FRAME_WIDTH,
3                          .FRAME_LENGTH,
3                          DEST_OFFSET);
3
3     FIND_SOURCE_PTR (    .SCB,
3                          SRCE_STR[0],
3                          CURRENT_PTR[0,0,.SCB[PSM$L_PAGE_WIDTH]],
3                          .FRAME_WIDTH,
3                          .DEST_OFFSET,
3                          SOURCE_OFFSET);
3
3 ! Exit loop when no source string
3 !
3     IF .SRCE_STR[SIZE] EQL 0
3     THEN
4         BEGIN
4         RET_LEN[0] = .FRAME_LENGTH - .I;
4         RETURN;
3         END;
3
3     ! Move the source to the destination
3     !
3     MOVE_FRAME (
3                 .SCB,
3                 SRCE_STR[0],                        ! string frame reference
3                 CURRENT_PTR[0,.DEST_OFFSET,.SCB[PSM$L_PAGE_WIDTH]],
3                                                     ! ref to frame
3                 .FRAME_WIDTH,                       ! cols to fill
3                 1);                                 ! rows to fill
3
3     ! Clear the source position
3     !
3     MOVE_FRAME (
3                 .SCB,
3                 CLR_STR[0],                         ! string frame reference
3                 CURRENT_PTR[0,.SOURCE_OFFSET,.SCB[PSM$L_PAGE_WIDTH]],
3                                                     ! ref to frame
3                 .FRAME_WIDTH,                       ! cols to fill
3                 1);                                 ! rows to fill
3
3     SRCE_STR[SIZE] = K_MAX_SIZE;
3
```

```
; 5702          6603  2      END;
; 5703          6604  1 END;



                                    007C 00000 MERGE_FRAME:
                                                      .WORD   Save R2,R3,R4,R5,R6                     ; 6517
                   5E   FEE8     CE   9E 00002        MOVAB   -280(SP), SP                            ;
                   50     08     AC   D0 00007        MOVL    CHAR_STRING, R0                         ; 6540
              FC   AD     04     A0   D0 0000B        MOVL    4(R0), CLR_STR+4                        ;
                   56     10     AC   D0 00010        MOVL    FRAME_WIDTH, R6                         ; 6541
              F8   AD            56   D0 00014        MOVL    R6, CLR_STR                             ;
56    20           6E            00   2C 00018        MOVC5   #0, (SP), #32, R6, @CLR_STR+4           ; 6542
                          FC     BD      0001D                                                        ;
              F0   AD   0100     8F   3C 0001F        MOVZWL  #256, SRCE_STR                          ; 6545
              F4   AD     08     AE   9E 00025        MOVAB   BUFFER, SRCE_STR+4                      ; 6546
                   53     0C     AC   D0 0002A        MOVL    FRAME_PTR, CURRENT_PTR                  ; 6550
      52      14   AC            01   C3 0002E        SUBL3   #1, FRAME_LENGTH, R2                    ; 6552
              04   AE            52   D0 00033        MOVL    R2, SOURCE_OFFSET                       ;
                   6E            52   D0 00037        MOVL    R2, DEST_OFFSET                         ; 6553
                   54     04     AC   D0 0003A        MOVL    SCB, R4                                 ; 6558
                                 5E   DD 0003E 1$:    PUSHL   SP                                      ; 6560
                          14     AC   DD 00040        PUSHL   FRAME_LENGTH                            ; 6562
                        0048     8F   BB 00043        PUSHR   #^M<R3,R6>                              ; 6560
                          F8     AD   9F 00047        PUSHAB  CLR_STR                                 ; 6559
                                 54   DD 0004A        PUSHL   R4                                      ; 6560
           0000V   CF            06   FB 0004C        CALLS   #6, FIND_DEST_PTR                       ;
                          04     AE   9F 00051        PUSHAB  SOURCE_OFFSET                           ; 6567
                          04     AE   DD 00054        PUSHL   DEST_OFFSET                             ; 6569
                        0048     8F   BB 00057        PUSHR   #^M<R3,R6>                              ; 6567
                          F0     AD   9F 0005B        PUSHAB  SRCE_STR                                ; 6566
                                 54   DD 0005E        PUSHL   R4                                      ; 6567
           0000V   CF            06   FB 00060        CALLS   #6, FIND_SOURCE_PTR                     ;
                          F0     AD   D5 00065        TSTL    SRCE_STR                                ; 6574
                                 07   12 00068        BNEQ    2$                                      ;
18    BC      14   AC            52   A3 0006A        SUBW3   I, FRAME_LENGTH, @RET_LEN               ; 6577
                                      04 00070        RET                                             ; 6576
                                 01   DD 00071 2$:    PUSHL   #1                                      ; 6586
                                 56   DD 00073        PUSHL   R6                                      ; 6588
      50      08   AE   0200     C4   C5 00075        MULL3   512(R4), DEST_OFFSET, R0                ; 6586
                               6043   9F 0007C        PUSHAB  (R0)[CURRENT_PTR]                       ;
                          F0     AD   9F 0007F        PUSHAB  SRCE_STR                                ; 6585
                                 54   DD 00082        PUSHL   R4                                      ; 6586
            FCFF   CF            05   FB 00084        CALLS   #5, MOVE_FRAME                          ;
                                 01   DD 00089        PUSHL   #1                                      ; 6596
                                 56   DD 0008B        PUSHL   R6                                      ; 6598
      50      0C   AE   0200     C4   C5 0008D        MULL3   512(R4), SOURCE_OFFSET, R0              ; 6596
                               6043   9F 00094        PUSHAB  (R0)[CURRENT_PTR]                       ;
                          F8     AD   9F 00097        PUSHAB  CLR_STR                                 ; 6595
                                 54   DD 0009A        PUSHL   R4                                      ; 6596
            FCE7   CF            05   FB 0009C        CALLS   #5, MOVE_FRAME                          ;
              F0   AD   0100     8F   3C 000A1        MOVZWL  #256, SRCE_STR                          ; 6601
                                 52   D7 000A7        DECL    I                                       ; 6555
                                 93   11 000A9        BRB     1$                                      ;
```

```
; Routine Size:  171 bytes,    Routine Base:  CODE + 3025
```

```
1 %sbttl 'INSERT_NAME_BANNER - Insert User Name as Banner into this Frame'
1 !++
1 ! Functional Description:
1 !       This procedure inserts a string into the cneter of an array(frame).
1 !       The workhorse of this routine is the BANNER routine which creates the
1 !       large letters.  Insertion is attempted in the center of the frame.
1 !       Otherwise, insertion starts at the left margin until no more characters
1 !       will fit.
1 !       NOTE:   Returns the amount of space used in the frame
1 !
1 ! Formal Parameters:
1 !       SCB                         - Address of the SCB
1 !       CHAR_DESC                   - Desc String to Insert
1 !       FRAME_PTR                   - Address of first byte of Frame
1 !       FRAME_LENGTH                - Length of Frame and Largest Banner
1 !       FRAME_WIDTH                 - Width of Frame and Height of Characters
1 !       DESIRED_BAN_TYPE            - Requested banner type
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               Truncation is possible.
1 !
1 !--
1 ROUTINE INSERT_NAME_BANNER (
1       SCB             : REF $BBLOCK,
1       CHAR_DESC       : REF VECTOR[2],
1       FRAME_PTR       : REF PAGE_ARRAY,
1       FRAME_WIDTH     ,                       ! Number of Columns
1       FRAME_LENGTH    ,                       ! Number of Rows
1       DESIRED_BAN_TYPE                        ! Banner size desired
1       )  =
2 BEGIN
2 !  Define literals to use in "Banner" Call
2 !  (incl... char_repeat, line_repeat, spacing)
2 !
2 LITERAL K_LARGE_LETTERS =  14,                ! Double size chars
2       K_MAX_STRING_SIZE = 42,                 ! max expanded chars(512 buffer)
2       K_ALT_CHAR      =   0,                  ! alternate construction char
2       K_SPACING       =   2,                  ! between character spacing
2       K_LEAD_SPACES   =   0,                  ! number of leading spaces
2       K_MAX_BUF       = 512,                  ! max for this frame buffer
2       LEAD_MASK       = %B'00100000',         ! convert lower to upper case
2       TRAILING        =   1;                  ! flag for discard
2                                               !   (anything but 0 is trailing)
2 !
2 !  By defining local buffer and descriptor. I can call the banner routine
2 !  and get the length of the string ... then use an algorithm to center the
2 !  string into the frame.
2 !
```

```
2 LOCAL
2     NUM_LINES    ,
2     CHAR_REPEAT  ,                                  ! character repeat
2     LINE_REPEAT  ,                                  ! line repeat
2     BUFFER       : VECTOR [512,byte],               ! assume max size 512 bytes
2     STRING_DESC  : VECTOR [2],                      ! descriptor to current string
2     STR_PTR      ,                                  ! temp addr of input string
2     STR_LEN      ,                                  ! temp length of input string
2     RET_LEN      : VECTOR[1];                       ! Return Length Used
2
2 ! Dont even try if no frame
2 !
3 IF (.FRAME_LENGTH LSS 7)                            ! won't ever fit !!
2       OR
3    (.FRAME_WIDTH LEQ 0)                             ! nadda...
2 THEN
2     RETURN 0;                                       ! dont even try... no room...
2
2 STR_LEN = .CHAR_DESC[SIZE];                         ! move them into registers
2 STR_PTR = .CHAR_DESC[ADDR];
2
2 BAS$EDIT (CHAR_DESC[0], CHAR_DESC[0], UPCASE_MASK);
2                                                     ! lower to upper case character
2 ! Insert only the string ... No trailing blanks
2 !
2 DISCARD (TRAILING, %C' ', .STR_PTR, .STR_LEN,
2          STR_LEN, STR_PTR);                         ! Return length and pointer
2
2 !
2 !  init the character spacing ...depends on frame_length passed in !
2 !       assume small chars.
2 CHAR_REPEAT =    1;                                 ! times to repeat a char
2 LINE_REPEAT =    1;                                 ! times to repeat a line
2 NUM_LINES =      7;                                 ! lines equal height of banner
2
2 ! Attempt to give the caller what he wants
2 !    Only use large banners if they fit in the Frame
2 !
3 IF (
4     (.DESIRED_BAN_TYPE EQL K_LARGE_LETTERS)         ! if he wants it
3             AND
5     ((.FRAME_LENGTH GEQ K_LARGE_LETTERS)            ! and...
4             AND                                     ! if there is room !!!
4     ((.FRAME_WIDTH/12) GEQ .STR_LEN))
3                                    )
2 THEN
3     BEGIN
3     CHAR_REPEAT =    2;                             ! times to repeat a char
3     LINE_REPEAT =    2;                             ! times to repeat a line
3     NUM_LINES   =    14;                            ! height of banner
2     END;
2
2 ! truncate long character names to fit in buffer
2 !
2 IF .STR_LEN GTR K_MAX_STRING_SIZE
2 THEN
2     STR_LEN = K_MAX_STRING_SIZE;                    ! maximum 42 chars in
```

```
; 5819      6719  2                                                        !   buffer of 512
; 5820      6720  2 ! get the buffer
; 5821      6721  2 !
; 5822      6722  2 STRING_DESC[SIZE] = %ALLOCATION(BUFFER);
; 5823      6723  2 STRING_DESC[ADDR] = BUFFER;
; 5824      6724  2
; 5825      6725  3 INCR LINE_NO FROM 0 TO (.NUM_LINES - 1)
; 5826      6726  2 DO
; 5827      6727  3      BEGIN
; 5828      6728  3      PSM$BANNER (.SCB,
; 5829      6729  3                  .STR_LEN,
; 5830      6730  3                  .STR_PTR,
; 5831      6731  3                  K_LEAD_SPACES,
; 5832      6732  3                  .CHAR_REPEAT,
; 5833      6733  3                  .LINE_REPEAT,
; 5834      6734  3                  K_SPACING,
; 5835      6735  3                  K_ALT_CHAR,
; 5836      6736  3                  .STRING_DESC[ADDR],
; 5837      6737  3                  .LINE_NO,
; 5838      6738  3                  STRING_DESC[SIZE]);
; 5839      6739  3
; 5840      6740  3      CENTER_FRAME ( .SCB,
; 5841      6741  3                      STRING_DESC[0],
; 5842      6742  3                      FRAME_PTR[0,.LINE_NO,.SCB[PSM$L_PAGE_WIDTH]],
; 5843      6743  3                      .FRAME_WIDTH, 1);
; 5844      6744  3      ! re-init
; 5845      6745  3      STRING_DESC[SIZE] = K_MAX_BUF;                  ! reset buffer size
; 5846      6746  2      END;
; 5847      6747  2
; 5848      6748  2 RETURN .NUM_LINES
; 5849      6749  1 END;



                                    007C 00000 INSERT_NAME_BANNER:
                                                       .WORD    Save R2,R3,R4,R5,R6                        ; 6636
                  5E      FDF0   CE   9E 00002         MOVAB    -528(SP), SP                               ;
                  07        14   AC   D1 00007         CMPL     FRAME_LENGTH, #7                           ; 6674
                                 05   19 0000B         BLSS     1$                                         ;
                            10   AC   D5 0000D         TSTL     FRAME_WIDTH                                ; 6676
                                 03   14 00010         BGTR     2$                                         ;
                               00BD   31 00012 1$:     BRW      7$                                         ;
                  50        08   AC   D0 00015 2$:     MOVL     CHAR_DESC, R0                              ; 6680
            04    AE             60   D0 00019         MOVL     (R0), STR_LEN                              ;
                  6E        04   A0   D0 0001D         MOVL     4(R0), STR_PTR                             ; 6681
                                 20   DD 00021         PUSHL    #32                                        ; 6683
                                 50   DD 00023         PUSHL    R0                                         ;
                                 50   DD 00025         PUSHL    R0                                         ;
     00000000G    00             03   FB 00027         CALLS    #3, BAS$EDIT                               ;
                                 5E   DD 0002E         PUSHL    SP                                         ; 6687
                            08   AE   9F 00030         PUSHAB   STR_LEN                                    ;
                            0C   AE   DD 00033         PUSHL    STR_LEN                                    ;
                            0C   AE   DD 00036         PUSHL    STR_PTR                                    ;
                                 20   DD 00039         PUSHL    #32                                        ;
                                 01   DD 0003B         PUSHL    #1                                         ;
```

```
             0000V   CF            06 FB 0003D        CALLS   #6, DISCARD                 ;
                     56            01 D0 00042        MOVL    #1, CHAR_REPEAT             ; 6693
                     55            01 D0 00045        MOVL    #1, LINE_REPEAT             ; 6694
                     54            07 D0 00048        MOVL    #7, NUM_LINES               ; 6695
                     0E       18   AC D1 0004B        CMPL    DESIRED_BAN_TYPE, #14       ; 6701
                                   1A 12 0004F        BNEQ    3$                          ;
                     0E       14   AC D1 00051        CMPL    FRAME_LENGTH, #14           ; 6703
                                   14 19 00055        BLSS    3$                          ;
    50          10   AC            0C C7 00057        DIVL3   #12, FRAME_WIDTH, R0        ; 6705
                04   AE            50 D1 0005C        CMPL    R0, STR_LEN                 ;
                                   09 19 00060        BLSS    3$                          ;
                     56            02 D0 00062        MOVL    #2, CHAR_REPEAT             ; 6709
                     55            02 D0 00065        MOVL    #2, LINE_REPEAT             ; 6710
                     54            0E D0 00068        MOVL    #14, NUM_LINES              ; 6711
                     2A       04   AE D1 0006B 3$:    CMPL    STR_LEN, #42                ; 6716
                                   04 15 0006F        BLEQ    4$                          ;
                04   AE            2A D0 00071        MOVL    #42, STR_LEN                ; 6718
                08   AE     0200   8F 3C 00075 4$:    MOVZWL  #512, STRING_DESC           ; 6722
                0C   AE       10   AE 9E 0007B        MOVAB   BUFFER, STRING_DESC+4       ; 6723
                     53       04   AC D0 00080        MOVL    SCB, R3                     ; 6742
                     52            01 CE 00084        MNEGL   #1, LINE_NO                 ;
                                   41 11 00087        BRB     6$                          ;
                              08   AE 9F 00089 5$:    PUSHAB  STRING_DESC                 ; 6738
                                   52 DD 0008C        PUSHL   LINE_NO                     ; 6737
                              14   AE DD 0008E        PUSHL   STRING_DESC+4               ; 6736
                     7E            02 7D 00091        MOVQ    #2, -(SP)                   ; 6728
                                   55 DD 00094        PUSHL   LINE_REPEAT                 ; 6733
                                   56 DD 00096        PUSHL   CHAR_REPEAT                 ; 6732
                                   7E D4 00098        CLRL    -(SP)                       ; 6728
                              20   AE DD 0009A        PUSHL   STR_PTR                     ; 6730
                              28   AE DD 0009D        PUSHL   STR_LEN                     ; 6729
                              04   AC DD 000A0        PUSHL   SCB                         ; 6728
         00000000G   00            0B FB 000A3        CALLS   #11, PSM$BANNER             ;
                                   01 DD 000AA        PUSHL   #1                          ; 6742
                              10   AC DD 000AC        PUSHL   FRAME_WIDTH                 ; 6743
    50               52     0200   C3 C5 000AF        MULL3   512(R3), LINE_NO, R0        ; 6742
                              0C BC40 9F 000B5        PUSHAB  @FRAME_PTR[R0]              ;
                              14   AE 9F 000B9        PUSHAB  STRING_DESC                 ; 6741
                              04   AC DD 000BC        PUSHL   SCB                         ; 6742
              FD63   CF            05 FB 000BF        CALLS   #5, CENTER_FRAME            ;
                08   AE     0200   8F 3C 000C4        MOVZWL  #512, STRING_DESC           ; 6745
    BB               52            54 F2 000CA 6$:    AOBLSS  NUM_LINES, LINE_NO, 5$      ; 6725
                     50            54 D0 000CE        MOVL    NUM_LINES, R0               ; 6748
                                      04 000D1        RET                                 ;
                                   50 D4 000D2 7$:    CLRL    R0                          ; 6749
                                      04 000D4        RET                                 ;

; Routine Size:  213 bytes,    Routine Base:  CODE + 30D0
```

```
1 %sbttl 'FIND_DEST_PTR - Finds an empty Position in the Frame'
1 !++
1 ! Functional Description:
1 !       This routine finds the first empty frame position from the bottom
1 !       of the frame.  The returned parameters include the dest_ptr (position
1 !       found) and the length left in the frame.  If unable to find an empty
1 !       position then RET_OFFSET = .FRAME_LENGTH
1 !
1 ! Formal Parameters:
1 !       SCB                          - Address of the SCB
1 !       CLR_STR                      - Descriptor of blank string
1 !       FRAME_PTR                    - Address of first byte of Frame
1 !       FRAME_LENGTH                 - Length of Frame and Largest Banner
1 !       FRAME_WIDTH                  - Width of Frame and Height of Characters
1 !       RETURN_PTR                   - Pointer to position in frame
1 !       RET_LEN                      - Resultant length of frame
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               Truncation is possible.
1 !
1 !--
1 ROUTINE FIND_DEST_PTR (
1         SCB             : REF $BBLOCK,
1         CLR_STR         : REF VECTOR[2],
1         FRAME_PTR       : REF PAGE_ARRAY,
1         FRAME_WIDTH     ,                 ! Number of Columns
1         FRAME_LENGTH    ,                 ! Number of Rows
1         RET_OFFSET      : REF VECTOR
1         )  :    NOVALUE =
2 BEGIN
2 LOCAL
2     CURR_PTR    ;
2
2 ! exit if frame length is zero
2 !
2 IF .FRAME_LENGTH EQL 0
2 THEN
2     RETURN;
2
2 DECR I FROM (.FRAME_LENGTH-1) TO 0
2 DO
3     BEGIN
3     CURR_PTR = FRAME_PTR[0, .I, .SCB[PSM$L_PAGE_WIDTH]];
3
3     IF CH$EQL( .FRAME_WIDTH, .CURR_PTR,
3                1, CH$PTR( UPLIT (' ')), %C' ' )
3     THEN
4         BEGIN
```

```
; 5908          6807  4            RET_OFFSET[0] = .I;
; 5909          6808  4            RETURN;                                  ! exit this routine
; 5910          6809  3            END;
; 5911          6810  2        END;
; 5912          6811  2
; 5913          6812  2 RET_OFFSET[0] = 0;                                  ! exit with 0 if destination not found
; 5914          6813  2
; 5915          6814  1 END;



                                                   031A5          .BLKB    3
                              00  00  00  20       031A8 P.AGN:   .ASCII   \ \<0><0><0>                                 ;



                                             007C 00000 FIND_DEST_PTR:
                                                                  .WORD    Save R2,R3,R4,R5,R6                          ; 6780
                                        14   AC  D5 00002         TSTL     FRAME_LENGTH                                 ; 6794
                                             2A  13 00005         BEQL     3$                                           ;
                              55        04   AC  D0 00007         MOVL     SCB, R5                                      ; 6801
                              54        14   AC  D0 0000B         MOVL     FRAME_LENGTH, I                              ; 6803
                                             1A  11 0000F         BRB      2$                                           ;
                    50        54      0200   C5  C5 00011 1$:     MULL3    512(R5), I, R0                               ; 6801
                    56        50        0C   AC  C1 00017         ADDL3    FRAME_PTR, R0, CURR_PTR                      ;
          01        20        66        10   AC  2D 0001C         CMPC5    FRAME_WIDTH, (CURR_PTR), #32, #1, P.AGN      ; 6803
                                        D8   AF     00022                                                               ;
                                             05  12 00024         BNEQ     2$                                           ;
                         18   BC             54  D0 00026         MOVL     I, @RET_OFFSET                               ; 6807
                                                 04 0002A         RET                                                   ; 6806
                              E3             54  F4 0002B 2$:     SOBGEQ   I, 1$                                        ; 6798
                                        18   BC  D4 0002E         CLRL     @RET_OFFSET                                  ; 6812
                                                 04 00031 3$:     RET                                                   ; 6814

; Routine Size:  50 bytes,    Routine Base:  CODE + 31AC
```

```
1 %sbttl 'FIND_SOURCE_PTR - Finds an empty Position in the Frame'
1 !++
1 ! Functional Description:
1 !         This routine finds the first nonempty frame position from the bottom
1 !         of the frame.  The returned parameters include the ret_str descriptor
1 !         return_ptr (position found) and the length left in the frame.  If
1 !         unable to find a string position then RET_STR[SIZE]=0, RET_LEN = 0 and
1 !         RETURN_PTR = FRAME_PTR.
1 !
1 ! Formal Parameters:
1 !         SCB                      - Address of the SCB
1 !         RET_STR                  - Descriptor of buffer for return string
1 !         FRAME_PTR                - Address of first byte of Frame
1 !         FRAME_LENGTH             - Length of Frame and Largest Banner
1 !         FRAME_WIDTH              - Width of Frame and Height of Characters
1 !         RET_OFFSET               - Pointer to position in frame
1 !
1 ! Implicit Inputs:
1 !                   none
1 !
1 ! Implicit Outputs:
1 !                   none
1 !
1 ! Returned Value:
1 !                   none
1 !
1 ! Side Effects:
1 !                   Truncation is possible.
1 !
1 !--
1 ROUTINE FIND_SOURCE_PTR (
1         SCB              : REF $BBLOCK,
1         RET_STR          : REF VECTOR[2],
1         FRAME_PTR        : REF PAGE_ARRAY,
1         FRAME_WIDTH      ,                  ! Number of Columns
1         FRAME_LENGTH     ,                  ! Number of Rows
1         RET_OFFSET       : REF VECTOR
1         )  :     NOVALUE =
2 BEGIN
2
2 LOCAL
2     CURR_PTR;
2
2
2 ! exit if frame length is zero
2 !
2 IF .FRAME_LENGTH EQL 0
2 THEN
3     BEGIN
3     RET_STR[SIZE] = 0;
3     RETURN;
2     END;
2
2 DECR I FROM (.FRAME_LENGTH-1) TO 0
2 DO
3     BEGIN
3     CURR_PTR = FRAME_PTR[0,.I,.SCB[PSM$L_PAGE_WIDTH]];
```

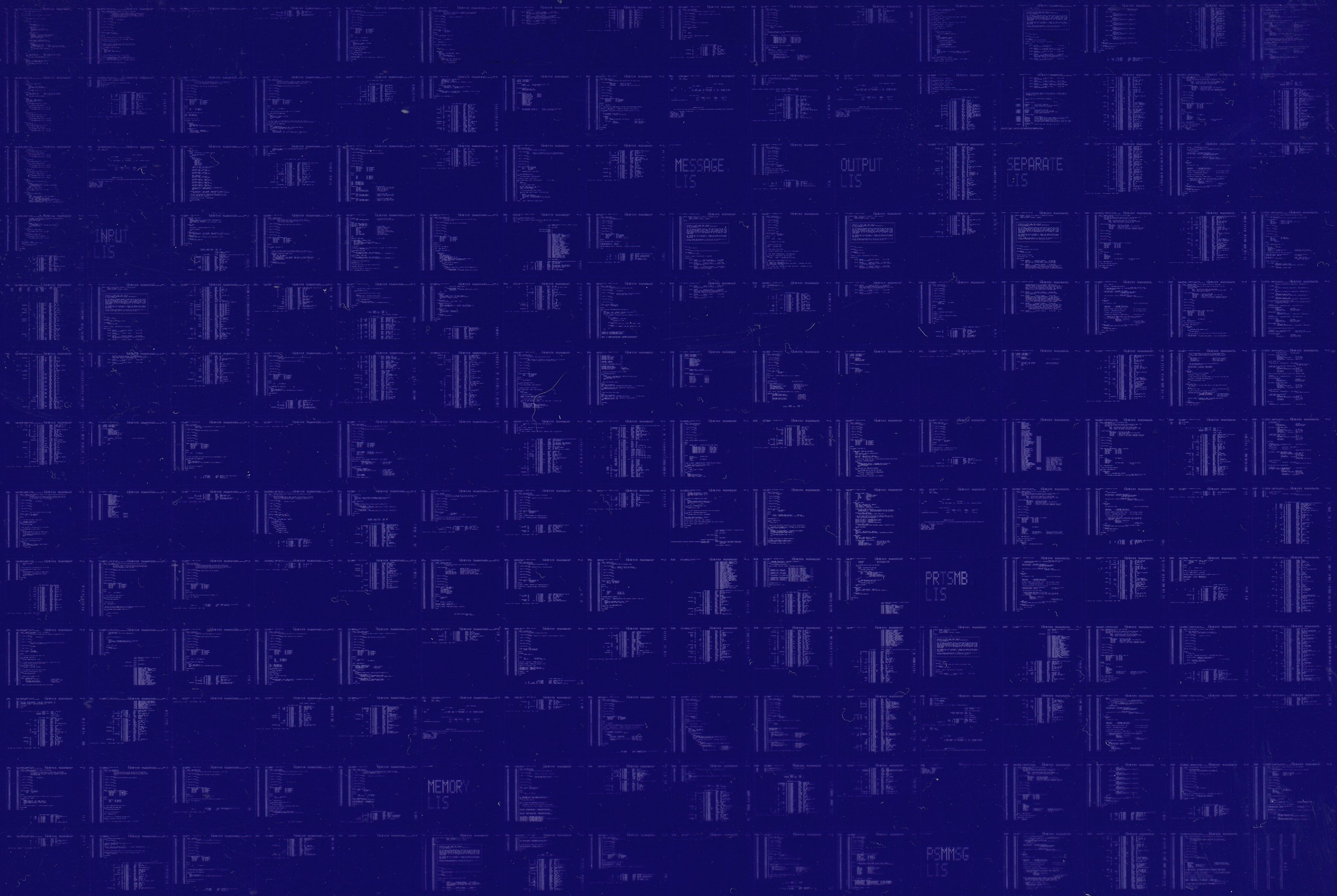
```
; 5974          6872 3
; 5975          6873 3        IF CH$NEQ( .FRAME_WIDTH, .CURR_PTR,
; 5976          6874 3                1, CH$PTR( UPLIT ('_')), %C' ' )
; 5977          6875 3        THEN
; 5978          6876 4            BEGIN
; 5979          6877 4            CH$MOVE(.FRAME_WIDTH, .CURR_PTR,
; 5980          6878 4                    .RET_STR[ADDR]);
; 5981          6879 4            RET_STR[SIZE] = .FRAME_WIDTH;
; 5982          6880 4            RET_OFFSET[0] = .I;
; 5983          6881 4            RETURN;                         ! exit this routine
; 5984          6882 3            END;
; 5985          6883 2        END;
; 5986          6884 2
; 5987          6885 2 RET_STR[SIZE] = 0;
; 5988          6886 2 RET_OFFSET = 0;
; 5989          6887 2
; 5990          6888 1 END;



                                                 031DE          .BLKB   2
                                 00  00  00  20  031E0 P.AGO:   .ASCII  \ \<0><0><0>                                      ;



                                            03FC 00000 FIND_SOURCE_PTR:
                                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9                      ; 6845
                                    14  AC  D5 00002            TSTL    FRAME_LENGTH                                      ; 6861
                                        04  12 00005            BNEQ    1$                                                ;
                                    08  BC  D4 00007            CLRL    @RET_STR                                          ; 6864
                                            04 0000A            RET                                                       ; 6863
                            58      04  AC  D0 0000B 1$:        MOVL    SCB, R8                                           ; 6871
                            56      14  AC  D0 0000F            MOVL    FRAME_LENGTH, I                                   ; 6873
                                        28  11 00013            BRB     3$                                                ;
                50          56    0200  C8  C5 00015 2$:        MULL3   512(R8), I, R0                                    ; 6871
                59          50      0C  AC  C1 0001B            ADDL3   FRAME_PTR, R0, CURR_PTR                           ;
      01        20          69      10  AC  2D 00020            CMPC5   FRAME_WIDTH, (CURR_PTR), #32, #1, P.AGO           ; 6873
                                    D4  AF     00026                                                                      ;
                                        13  13 00028            BEQL    3$                                                ;
                            57      08  AC  D0 0002A            MOVL    RET_STR, R7                                       ; 6878
             04  B7         69      10  AC  28 0002E            MOVC3   FRAME_WIDTH, (CURR_PTR), @4(R7)                   ;
                            67      10  AC  D0 00034            MOVL    FRAME_WIDTH, (R7)                                 ; 6879
                       18  BC           56  D0 00038            MOVL    I, @RET_OFFSET                                    ; 6880
                                            04 0003C            RET                                                       ; 6876
                           D5           56  F4 0003D 3$:        SOBGEQ  I, 2$                                             ; 6868
                                    08  BC  D4 00040            CLRL    @RET_STR                                          ; 6885
                                    18  AC  D4 00043            CLRL    RET_OFFSET                                        ; 6886
                                            04 00046            RET                                                       ; 6888

; Routine Size:  71 bytes,    Routine Base:  CODE + 31E4
; 5992          6889  1 %sbttl 'DELIMIT_STRING - Return the last position of this delimiter'
; 5993          6890  1 !++
; 5994          6891  1 ! Functional Description:
; 5995          6892  1 !       This procedure returns the position of the delimiter nearest the
; 5996          6893  1 !       string_end
; 5997          6894  1 !
; 5998          6895  1 ! Formal Parameters:
; 5999          6896  1 !       STR_PTR             - Pointer of String to delimit
; 6000          6897  1 !       CHAR                - Character delimiter
; 6001          6898  1 !       STR_END             - End position of string
; 6002          6899  1 !
; 6003          6900  1 ! Implicit Inputs:
; 6004          6901  1 !               none
; 6005          6902  1 !
; 6006          6903  1 ! Implicit Outputs:
; 6007          6904  1 !               none
; 6008          6905  1 !
; 6009          6906  1 ! Returned Value:
; 6010          6907  1 !               none
; 6011          6908  1 !
; 6012          6909  1 ! Side Effects:
; 6013          6910  1 !               none
; 6014          6911  1 !--
; 6015          6912  1 ROUTINE DELIMIT_STRING  (
; 6016          6913  1       STR_PTR ,
; 6017          6914  1       CHAR    ,
; 6018          6915  1       STR_END           )       =
; 6019          6916  2 BEGIN
; 6020          6917  2 LOCAL
; 6021          6918  2     POS,
; 6022          6919  2     BASE,
; 6023          6920  2     TEMP_PTR,
; 6024          6921  2     CHAR_PTR;
; 6025          6922  2
; 6026          6923  2 TEMP_PTR = .STR_PTR;
; 6027          6924  2 BASE = .STR_PTR + .STR_END - 1;
; 6028          6925  2 CHAR_PTR = CH$PTR(CHAR);
; 6029          6926  2
; 6030          6927  2 DECR CURR_PTR FROM (.BASE) TO .STR_PTR DO
; 6031          6928  3 BEGIN
; 6032          6929  3     TEMP_PTR = CH$PTR(.CURR_PTR);
; 6033          6930  3
; 6034          6931  3     POS = CH$EQL(1, .TEMP_PTR, 1, .CHAR_PTR);
; 6035          6932  3
; 6036          6933  3     IF  (.POS EQL 1) AND
; 6037          6934  4         (.CURR_PTR EQL .STR_PTR)
; 6038          6935  3     THEN
; 6039          6936  3         RETURN .STR_END;          ! ...return the original length
; 6040          6937  3
; 6041          6938  3     IF  (.POS EQL 1) AND
; 6042          6939  4         (.CURR_PTR GTR .STR_PTR)
; 6043          6940  3     THEN                                          ! char in string
; 6044          6941  3         RETURN (.CURR_PTR - .STR_PTR + 1);        !  ...return position plus one
; 6045          6942  3
; 6046          6943  3     IF (.POS EQL 0) AND (.CURR_PTR EQL .STR_PTR) THEN
; 6047          6944  3         RETURN .STR_END;                          ! handle extra decrem
; 6048          6945  2     END;
; 6049          6946  2
; 6050          6947  2 RETURN .STR_END;          ! ...return the original length
; 6051          6948  2
; 6052          6949  1 END;



                                                           003C 00000 DELIMIT_STRING:
                                                                              .WORD   Save R2,R3,R4,R5                        ; 6912
                                         52        04    AC  D0 00002         MOVL    STR_PTR, R2                             ; 6923
                                         54              52  D0 00006         MOVL    R2, TEMP_PTR                            ;
                          50             52        0C    AC  C1 00009         ADDL3   STR_END, R2, R0                         ; 6924
                                                         50  D7 0000E         DECL    BASE                                    ;
                                         55        08    AC  9E 00010         MOVAB   CHAR, CHAR_PTR                          ; 6925
                                                         3A  11 00014         BRB     6$                                      ; 6931
                                         54              50  D0 00016 1$:     MOVL    CURR_PTR, TEMP_PTR                      ; 6929
                                                         51  D4 00019         CLRL    R1                                      ; 6931
                                         65              64  91 0001B         CMPB    (TEMP_PTR), (CHAR_PTR)                  ;
                                                         02  12 0001E         BNEQ    2$                                      ;
                                                         51  D6 00020         INCL    R1                                      ;
                                         53              51  D0 00022 2$:     MOVL    R1, POS                                 ;
                                                         51  D4 00025         CLRL    R1                                      ; 6933
                                         01              53  D1 00027         CMPL    POS, #1                                 ;
                                                         07  12 0002A         BNEQ    3$                                      ;
                                                         51  D6 0002C         INCL    R1                                      ;
                                         52              50  D1 0002E         CMPL    CURR_PTR, R2                            ; 6934
                                                         22  13 00031         BEQL    7$                                      ;
                                         0F              51  E9 00033 3$:     BLBC    R1, 4$                                  ; 6938
                                         52              50  D1 00036         CMPL    CURR_PTR, R2                            ; 6939
                                                         0A  15 00039         BLEQ    4$                                      ;
                          51             50              52  C3 0003B         SUBL3   R2, CURR_PTR, R1                        ; 6941
                                                         51  D6 0003F         INCL    R1                                      ;
                                         50              51  D0 00041         MOVL    R1, R0                                  ;
                                                             04 00044         RET                                             ;
                                                         53  D5 00045 4$:     TSTL    POS                                     ; 6943
                                                         05  12 00047         BNEQ    5$                                      ;
                                         52              50  D1 00049         CMPL    CURR_PTR, R2                            ;
                                                         07  13 0004C         BEQL    7$                                      ;
                                                         50  D7 0004E 5$:     DECL    CURR_PTR                                ; 6927
                                         52              50  D1 00050 6$:     CMPL    CURR_PTR, R2                            ;
                                                         C1  18 00053         BGEQ    1$                                      ;
                                         50        0C    AC  D0 00055 7$:     MOVL    STR_END, R0                             ; 6947
                                                             04 00059         RET                                             ; 6949

; Routine Size:  90 bytes,    Routine Base:  CODE + 322B
```

```
1 %sbttl 'DELIMIT_STRING_NOT - Return the last position of not this delimiter'
1 !++
1 ! Functional Description:
1 !       This procedure returns the length of the string without the delimited
1 !       characters on the string end.  Return the original length if
1 !       non_delimiters cannot be found.
1 !
1 ! Formal Parameters:
1 !       STR_PTR         - Pointer of String to delimit
1 !       CHAR            - Character delimiter
1 !       STR_END         - End position of string
1 !
1 ! Implicit Inputs:
1 !               none
1 !
1 ! Implicit Outputs:
1 !               none
1 !
1 ! Returned Value:
1 !               none
1 !
1 ! Side Effects:
1 !               none
1 !--
1 ROUTINE DELIMIT_STRING_NOT(
1         STR_PTR  ,
1         CHAR     ,
1         STR_END          )       =
2 BEGIN
2
2
2 LOCAL
2     PTR : REF VECTOR[,byte];
2
2 IF .STR_END EQL 0
2 THEN
2     RETURN 0;
2
2 PTR = .STR_PTR + .STR_END - 1;
2
2 WHILE .PTR GTRU .STR_PTR
2 DO
2     IF .PTR[0] NEQU .CHAR
2     THEN
2         EXITLOOP
2     ELSE
2         PTR = .PTR - 1;
2
2 RETURN .PTR - .STR_PTR + 1;
2
1 END;



                                    0000 00000 DELIMIT_STRING_NOT:
```

```
                                                             .WORD   Save nothing                        ; 6974
                                       0C   AC  D5 00002      TSTL    STR_END                             ; 6984
                                            1D  13 00005      BEQL    3$                                  ;
                        50       04   AC    0C   AC  C1 00007      ADDL3   STR_END, STR_PTR, R0                ; 6988
                                            50  D7 0000D 1$:  DECL    PTR                                 ;
                                 04   AC         50  D1 0000F      CMPL    PTR, STR_PTR                        ; 6990
                                            08  1B 00013      BLEQU   2$                                  ;
      08   AC           60            08         00  ED 00015      CMPZV   #0, #8, (PTR), CHAR                 ; 6992
                                            F0  13 0001B      BEQL    1$                                  ;
                                      50    04   AC  C2 0001D 2$:  SUBL2   STR_PTR, R0                         ; 6998
                                            50  D6 00021      INCL    R0                                  ;
                                                04 00023      RET                                         ;
                                            50  D4 00024 3$:  CLRL    R0                                  ; 7000
                                                04 00026      RET                                         ;

; Routine Size:  39 bytes,    Routine Base:  CODE + 3285
; 6106          7001   1 %sbttl 'DISCARD - Returns a Pointer to First Char NOT Discarded'
; 6107          7002   1 !++
; 6108          7003   1 ! Functional Description:
; 6109          7004   1 !       This routine discards the character in the string from the beginning
; 6110          7005   1 !       (LEADING) or end(TRAILING) of the string(STR_PTR) and returns a pointer
; 6111          7006   1 !       to the first position that is found not to contain the discard
; 6112          7007   1 !       character(CHAR).  The boundaries of the string are the beginning pointer
; 6113          7008   1 !       and the string length(LEN).
; 6114          7009   1 !
; 6115          7010   1 ! Formal Parameters:
; 6116          7011   1 !       WHICH_WAY         - Leading/Trailing
; 6117          7012   1 !       CHAR              - Character to discard
; 6118          7013   1 !       STR_PTR           - Pointer of String to delimit
; 6119          7014   1 !       LEN               - Length of string
; 6120          7015   1 !       RET_PTR           - Return pointer to first undiscarded position
; 6121          7016   1 !
; 6122          7017   1 ! Implicit Inputs:
; 6123          7018   1 !               none
; 6124          7019   1 !
; 6125          7020   1 ! Implicit Outputs:
; 6126          7021   1 !               none
; 6127          7022   1 !
; 6128          7023   1 ! Returned Value:
; 6129          7024   1 !               none
; 6130          7025   1 !
; 6131          7026   1 ! Side Effects:
; 6132          7027   1 !               none
; 6133          7028   1 !--
; 6134          7029   1 ROUTINE DISCARD (
; 6135          7030   1         WHICH_WAY,
; 6136          7031   1         CHAR,
; 6137          7032   1         STR_PTR ,
; 6138          7033   1         LEN     : WORD,                 ! force word size
; 6139          7034   1         RET_LEN : REF VECTOR,
; 6140          7035   1         RET_PTR : REF VECTOR) : NOVALUE =
; 6141          7036   2 BEGIN
; 6142          7037   2
; 6143          7038   2 LITERAL
; 6144          7039   2         LEADING    = 0;
; 6145          7040   2
; 6146          7041   2 LOCAL
; 6147          7042   2     DSTR_DESC   : VECTOR[2];
; 6148          7043   2
; 6149          7044   2 ! a descriptor is needed for BAS$EDIT routine
; 6150          7045   2 DSTR_DESC[SIZE] = .LEN;
; 6151          7046   2 DSTR_DESC[ADDR] = .STR_PTR;
; 6152          7047   2
; 6153          7048   2 IF .WHICH_WAY EQL LEADING THEN
; 6154          7049   3     BEGIN
; 6155          7050   3     BAS$EDIT ( DSTR_DESC[0], DSTR_DESC[0], LEAD_MASK);  ! trim leading blanks
; 6156          7051   3     RET_PTR[0] = .DSTR_DESC[ADDR];
; 6157          7052   3     RET_LEN[0] = .DSTR_DESC[SIZE];
; 6158          7053   3     END
; 6159          7054   2 ELSE                                                 ! trim trailing blanks
; 6160          7055   3     BEGIN
; 6161          7056   3     RET_LEN[0] = DELIMIT_STRING_NOT ( .STR_PTR, .CHAR, .LEN);
; 6162          7057   3     RET_PTR[0] = .STR_PTR;
; 6163          7058  2     END;
; 6164          7059  1 END;



                                        0000 00000 DISCARD:.WORD   Save nothing                                          ; 7029
                 5E                04   C2 00002           SUBL2   #4, SP                                                :
                 7E          10    AC   3C 00005           MOVZWL  LEN, DSTR_DESC                                        ; 7045
          04     AE          0C    AC   D0 00009           MOVL    STR_PTR, DSTR_DESC+4                                  ; 7046
                             04    AC   D5 0000E           TSTL    WHICH_WAY                                             ; 7048
                                   19   12 00011           BNEQ    1$                                                    :
                                   08   DD 00013           PUSHL   #8                                                    ; 7050
                             04    AE   9F 00015           PUSHAB  DSTR_DESC                                             :
                             08    AE   9F 00018           PUSHAB  DSTR_DESC                                             :
   00000000G     00                03   FB 0001B           CALLS   #3, BAS$EDIT                                          :
          18     BC          04    AE   D0 00022           MOVL    DSTR_DESC+4, @RET_PTR                                 ; 7051
          14     BC                6E   D0 00027           MOVL    DSTR_DESC, @RET_LEN                                   ; 7052
                                   04      0002B           RET                                                           ; 7048
                 7E          10    AC   3C 0002C 1$:       MOVZWL  LEN, -(SP)                                            ; 7056
                             08    AC   DD 00030           PUSHL   CHAR                                                  :
                             0C    AC   DD 00033           PUSHL   STR_PTR                                               :
          9F     AF                03   FB 00036           CALLS   #3, DELIMIT_STRING_NOT                                :
          14     BC                50   D0 0003A           MOVL    R0, @RET_LEN                                          :
          18     BC          0C    AC   D0 0003E           MOVL    STR_PTR, @RET_PTR                                     ; 7057
                                   04      00043           RET                                                           ; 7059

; Routine Size:  68 bytes,    Routine Base:  CODE + 32AC
; 6166            7060  1 %sbttl 'FILE_OPEN - Boolean Valued routine indicating file open status'
; 6167            7061  1 !++
; 6168            7062  1 ! Functional Description:
; 6169            7063  1 !       This routine interrogates the FAB and determines if the current file
; 6170            7064  1 !       is open and/or if information can be extracted from the file.
; 6171            7065  1 !       TRUE = 1, FALSE = 0;
; 6172            7066  1 !
; 6173            7067  1 ! Formal Parameters:
; 6174            7068  1 !               none
; 6175            7069  1 !
; 6176            7070  1 ! Implicit Inputs:
; 6177            7071  1 !               none
; 6178            7072  1 !
; 6179            7073  1 ! Implicit Outputs:
; 6180            7074  1 !               none
; 6181            7075  1 !
; 6182            7076  1 ! Returned Value:
; 6183            7077  1 !               none
; 6184            7078  1 !
; 6185            7079  1 ! Side Effects:
; 6186            7080  1 !               none
; 6187            7081  1 !--
; 6188            7082  1 ROUTINE FILE_OPEN (
; 6189            7083  1               SCB                     : REF $BBLOCK   ! SCB
; 6190            7084  1                )       =
; 6191            7085  2 BEGIN
; 6192            7086  2
; 6193            7087  2 RETURN .SCB[PSM$V_FAB_VALID]
; 6194            7088  2
; 6195            7089  1 END;



                                                0000 00000 FILE_OPEN:
                                                                 .WORD   Save nothing                                   ; 7082
                                    50      04   AC  D0 00002    MOVL    SCB, R0                                        ; 7087
         50       10   A0           01           04  EF 00006    EXTZV   #4, #1, 16(R0), R0                             ;
                                                     04 0000C    RET                                                    ; 7089

; Routine Size:  13 bytes,    Routine Base:  CODE + 32F0
; 6197          7090  1 END
; 6198          7091  0 ELUDOM




;                                       PSECT SUMMARY

;       Name                        Bytes                             Attributes

;    DATA                                  4  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;    CODE                              13053  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                               Library Statistics

;                                                  -------- Symbols --------          Pages        Processing
;       File                                       Total    Loaded    Percent          Mapped       Time

;    _$255$DUA28:[SYSLIB]LIB.L32;1                 18619        88          0           1000          00:01.9




;                                       COMMAND QUALIFIERS

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

; Size:           10301 code + 2756 data bytes
; Run Time:          03:23.1
; Elapsed Time:      06:52.7
; Lines/CPU Min:      2094
; Lexemes/CPU-Min: 22941
; Memory Used:  682 pages
; Compilation Complete
```

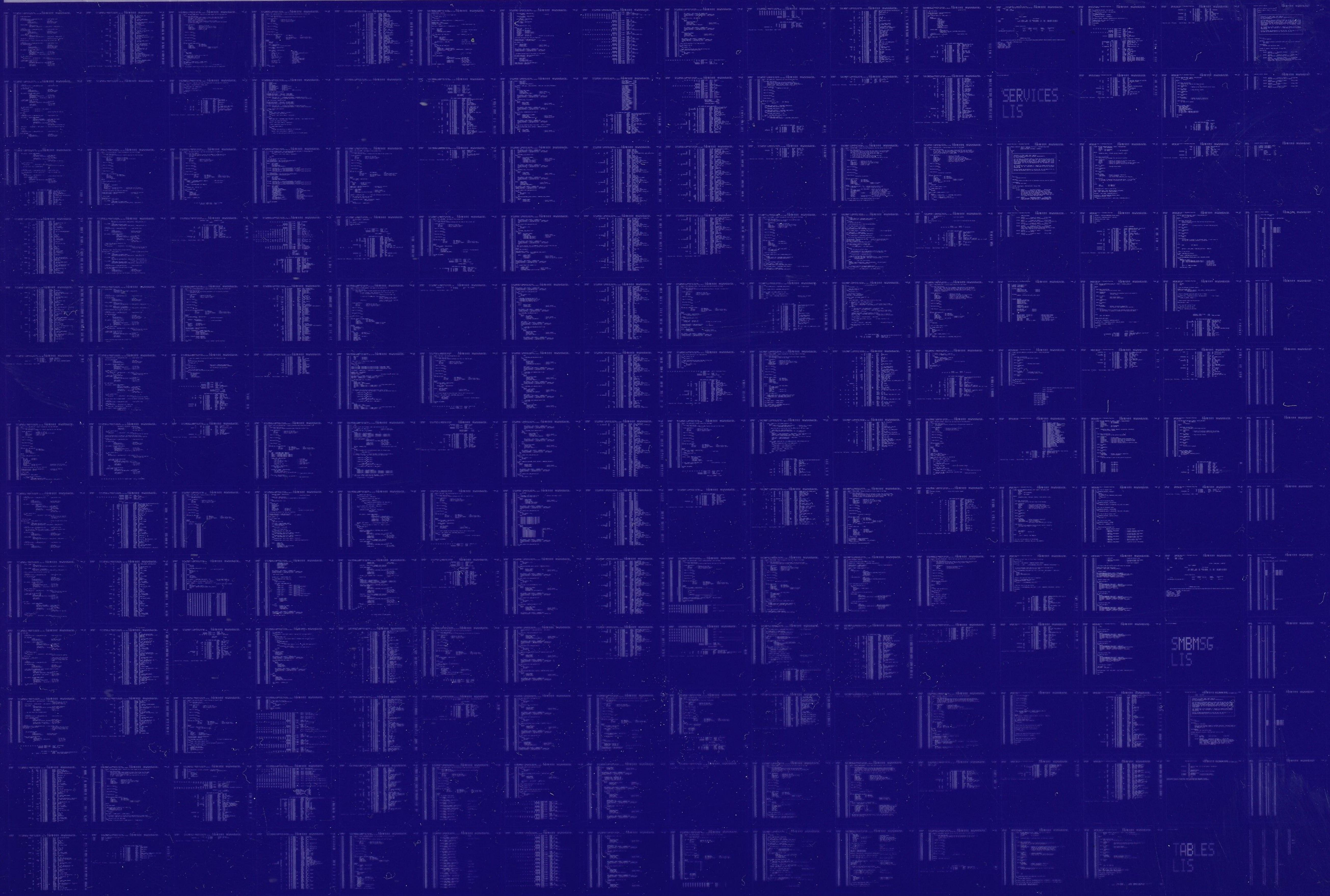
0310 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



0311 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



SERVICES
LIS

SMBMSG
LIS

TABLES
LIS