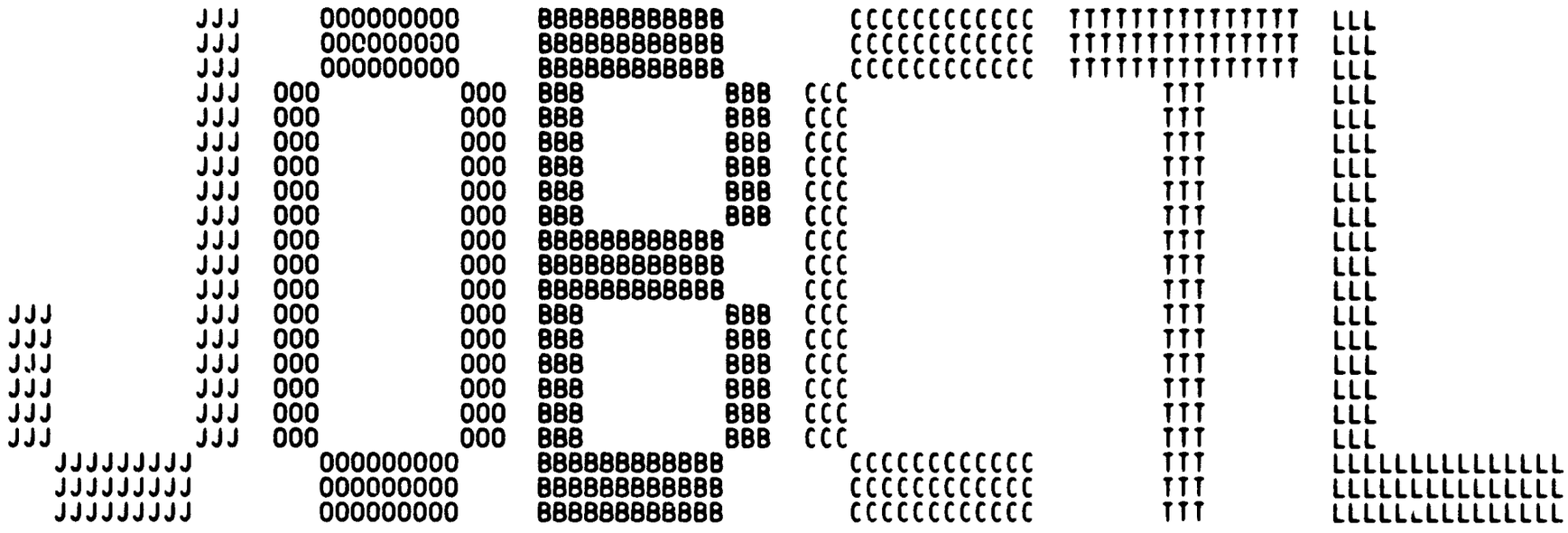

_$2

Val

0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
314
485
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FFF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF

FILEID**INITQUEUE

N 14

IN
VO

```
IIIIII    NN       NN    IIIIII    TTTTTTTTTT    QQQQQQ    UU      UU  EEEEEEEEEE  UU      UU  EEEEEEEEEE
IIIIII    NN       NN    IIIIII    TTTTTTTTTT    QQQQQQ    UU      UU  EEEEEEEEEE  UU      UU  EEEEEEEEEE
  II      NN       NN      II          TT       QQ    QQ  UU      UU  EE          UU      UU  EE
  II      NN       NN      II          TT       QQ    QQ  UU      UU  EE          UU      UU  EE
  II      NNNN     NN      II          TT       QQ    QQ  UU      UU  EE          UU      UU  EE
  II      NNNN     NN      II          TT       QQ    QQ  UU      UU  EE          UU      UU  EE
  II      NN NN    NN      II          TT       QQ    QQ  UU      UU  EEEEEEEE    UU      UU  EEEEEEEE
  II      NN NN    NN      II          TT       QQ    QQ  UU      UU  EEEEEEEE    UU      UU  EEEEEEEE
  II      NN  NNNN         II          TT       QQ QQ QQ  UU      UU  EE          UU      UU  EE
  II      NN  NNNN         II          TT       QQ QQ QQ  UU      UU  EE          UU      UU  EE
  II      NN       NN      II          TT       QQ  QQ    UU      UU  EE          UU      UU  EE              ....
  II      NN       NN      II          TT       QQ  QQ    UU      UU  EE          UU      UU  EE              ....
IIIIII    NN       NN    IIIIII        TT         QQQQ QQ  UUUUUUUUUU  EEEEEEEEEE  UUUUUUUUUU  EEEEEEEEEE  ....
IIIIII    NN       NN    IIIIII        TT         QQQQ QQ  UUUUUUUUUU  EEEEEEEEEE  UUUUUUUUUU  EEEEEEEEEE  ....

LL            IIIIII      SSSSSSS
LL            IIIIII      SSSSSSS                                                                          ;
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    SSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSS
```

```
;     1          0001  0 MODULE INITQUEUE(%TITLE 'Initialize system job queue file'
;     2          0002  0                    IDENT = 'V04-001'
;     3          0003  0                    ) =
;     4          0004  1 BEGIN
;     5          0005  1
;     6          0006  1 !
;     7          0007  1 !**********************************************************************
;     8          0008  1 !*                                                                    *
;     9          0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                          *
;    10          0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.           *
;    11          0011  1 !*  ALL RIGHTS RESERVED.                                           *
;    12          0012  1 !*                                                                    *
;    13          0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    14          0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    15          0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    16          0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    17          0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    18          0018  1 !*  TRANSFERRED.                                                   *
;    19          0019  1 !*                                                                    *
;    20          0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    21          0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    22          0022  1 !*  CORPORATION.                                                   *
;    23          0023  1 !*                                                                    *
;    24          0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    25          0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.         *
;    26          0026  1 !*                                                                    *
;    27          0027  1 !*                                                                    *
;    28          0028  1 !**********************************************************************
;    29          0029  1 !
;    30          0030  1
;    31          0031  1 !++
;    32          0032  1 ! FACILITY:
;    33          0033  1 !       Job controller.
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !       This module contains the routines that initialize and validate the
;    37          0037  1 !       system job queue file.
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT:
;    40          0040  1 !       VAX/VMS user and kernel mode.
;    41          0041  1 !--
;    42          0042  1 !
;    43          0043  1 ! AUTHOR: M. Jack, CREATION DATE: 16-Feb-1982
;    44          0044  1 !
;    45          0045  1 ! MODIFIED BY:
;    46          0046  1 !
;    47          0047  1 !
;    48          0048  1 !       V04-001 JAK0236         J A Krycka      14-Sep-1984
;    49          0049  1 !               Collect more diagnostic information.
;    50          0050  1 !
;    51          0051  1 !       V03-018 KPL0001         P Lieberwirth,  22-Jul-1984
;    52          0052  1 !               Add COLD_START_CURRENT_LIST routine to detect and repair
;    53          0053  1 !               SMQ current list corruption.  Rework WARM_START_EXISTING_FILE
;    54          0054  1 !               to be tolerant of system failures.
;    55          0055  1 !
;    56          0056  1 !       V03-017 JAK0221         J A Krycka      16-Jul-1984
;    57          0057  1 !               Always open the queue file shared (retract V03-013) because
```

```
;    58          0058  1 !                 non-shared access effectively disables use of the process-local
;    59          0059  1 !                 I/O buffer cache.
;    60          0060  1 !
;    61          0061  1 !       V03-016 JAK0219         J A Krycka      13-Jul-1984
;    62          0062  1 !                 Use the SJC$_BUFFER_COUNT and SJC$_EXTEND_QUANTITY items for the
;    63          0063  1 !                 SJC$_START_QUEUE_MANAGER function to override default values for
;    64          0064  1 !                 the MBF and ALQ fields during RMS operations on the queue file.
;    65          0065  1 !                 (These correspond to the /BUFFER_COUNT and /EXTEND_QUANTITY
;    66          0066  1 !                 qualifiers on the START/QUEUE/MANAGER command.)
;    67          0067  1 !
;    68          0068  1 !       V03-015 JAK0213         J A Krycka      18-May-1984
;    69          0069  1 !                 Use newly created LCK$M_NODLCKBLK (no deadlock on blocking AST)
;    70          0070  1 !                 option on enqueue service calls that specify a blocking AST
;    71          0071  1 !                 routine that will immediately relinquish the lock.
;    72          0072  1 !
;    73          0073  1 !       V03-014 JAK0212         J A Krycka      15-May-1984
;    74          0074  1 !                 Examine timer queue on warm start of the job queue file.
;    75          0075  1 !
;    76          0076  1 !       V03-013 JAK0211         J A Krycka      14-May-1984
;    77          0077  1 !                 Continuation of V03-012.
;    78          0078  1 !
;    79          0079  1 !       V03-012 JAK0210         J A Krycka      10-May-1984
;    80          0080  1 !                 If this node is not part of a cluster or the job queue file is
;    81          0081  1 !                 not on a cluster-wide accessible device, then reopen the queue
;    82          0082  1 !                 file for non-shared access (SHR=UPI) to avoid having RMS take
;    83          0083  1 !                 out locks on records in the local buffer cache.
;    84          0084  1 !
;    85          0085  1 !       V03-011 JAK0209         J A Krycka      08-May-1984
;    86          0086  1 !                 Display JBC$_STRUCT_LEVEL message if structure level of existing
;    87          0087  1 !                 queue file is incompatible with job controller implementation.
;    88          0088  1 !                 Also, log diagnostic information in DIAG_TRACE vector.
;    89          0089  1 !
;    90          0090  1 !       V03-010 JAK0202         J A Krycka      16-Apr-1984
;    91          0091  1 !                 Update selected queue file creation parameters.
;    92          0092  1 !
;    93          0093  1 !       V03-009 JAK0201         J A Krycka      10-Apr-1984
;    94          0094  1 !                 Use LCK$M_NODLCKWT (no deadlock wait) option on enqueue service
;    95          0095  1 !                 for the master queue file lock to avoid having the lock manager
;    96          0096  1 !                 declare a deadlock situation.
;    97          0097  1 !
;    98          0098  1 !       V03-008 MLJ0118         Martin L. Jack, 23-Aug-1983
;    99          0099  1 !                 Implement page setup.
;   100          0100  1 !
;   101          0101  1 !       V03-007 MLJ0115         Martin L. Jack, 30-Jul-1983
;   102          0102  1 !                 Changes for job controller baselevel.
;   103          0103  1 !
;   104          0104  1 !       V03-006 MLJ0114         Martin L. Jack, 23-Jun-1983
;   105          0105  1 !                 Changes for job controller baselevel.
;   106          0106  1 !
;   107          0107  1 !       V03-005 MLJ0113         Martin L. Jack, 26-May-1983
;   108          0108  1 !                 Changes for job controller baselevel.
;   109          0109  1 !
;   110          0110  1 !       V03-004 MLJ0112         Martin L. Jack, 29-Apr-1983
;   111          0111  1 !                 Changes for job controller baselevel.
;   112          0112  1 !
;   113          0113  1 !       V03-003 MLJ0111         Martin L. Jack, 20-Apr-1983
;   114          0114  1 !                 Correct another problem in warm start.
```

```
;  115          0115  1 !
;  116          0116  1 !       V03-002 MLJ0110         Martin L. Jack, 18-Apr-1983
;  117          0117  1 !               Correct problem in warm start and missing test for unwind.
;  118          0118  1 !
;  119          0119  1 !       V03-001 MLJ0109         Martin L. Jack, 14-Apr-1983
;  120          0120  1 !               Changes for job controller baselevel.
;  121          0121  1 !
;  122          0122  1 !**
```

```
;  124     0123  1 REQUIRE 'SRC$:JOBCTLDEF';
;  125     1164  1
;  126     1165  1
;  127     1166  1 LINKAGE
;  128     1167  1         L_TREE = CALL: GLOBAL(RTM=10, SQH=11);
;  129     1168  1
;  130     1169  1
;  131     1170  1 FORWARD ROUTINE
;  132     1171  1         REFERENCE_ENTRY_NUMBER:          L_TREE NOVALUE,
;  133     1172  1         ENQUEUE:                         L_TREE NOVALUE,
;  134     1173  1         TREE:                            L_TREE NOVALUE,
;  135     1174  1         COLD_START_CURRENT_LIST:         NOVALUE,
;  136     1175  1         COLD_START_EXISTING_FILE:        NOVALUE,
;  137     1176  1         COLD_START_NEW_FILE:             NOVALUE,
;  138     1177  1         WARM_START_EXISTING_FILE:        NOVALUE,
;  139     1178  1         INITIALIZE_QUEUE_FILE_HANDLER,
;  140     1179  1         INITIALIZE_QUEUE_FILE;
;  141     1180  1
;  142     1181  1
;  143     1182  1 EXTERNAL ROUTINE
;  144     1183  1         AFTER_AST:                       NOVALUE,
;  145     1184  1         COMPLETE_JOB:                    NOVALUE,
;  146     1185  1         DEALLOCATE_MEMORY:               NOVALUE,
;  147     1186  1         DELETE_SJH_RECORD:               NOVALUE,
;  148     1187  1         ENQUEUE_JOB:                     L_OUTPUT_2 NOVALUE,
;  149     1188  1         FLUSH_RECORD:                    NOVALUE,
;  150     1189  1         JOB_SCHEDULING_POLICY,
;  151     1190  1         LOCK_QUEUE_FILE:                 NOVALUE,
;  152     1191  1         QUEUE_MASTER_AST:                NOVALUE,
;  153     1192  1         READ_RECORD,
;  154     1193  1         RELEASE_RECORD:                  NOVALUE,
;  155     1194  1         REMOTE_BLOCKING_AST:             NOVALUE,
;  156     1195  1         REWRITE_RECORD:                  NOVALUE,
;  157     1196  1         SCAN_INCOMPLETE_SERVICES:        NOVALUE,
;  158     1197  1         SIGNAL_FILE_ERROR:               NOVALUE,
;  159     1198  1         UPDATE_GETQOI_DATA:              NOVALUE,
;  160     1199  1         WRITE_ACCOUNTING_RECORD:         NOVALUE;
;  161     1200  1
;  162     1201  1
;  163     1202  1 BIND
;  164     1203  1
;  165     1204  1         ! KBF values for known block numbers.
;  166     1205  1         !
;  167     1206  1         SQH_KBF=        UPLIT(1),
;  168     1207  1         SFM_KBF=        UPLIT(2),       ! Initial form definition
;  169     1208  1         SFX_KBF=        UPLIT(3);       ! Initial form index block
;  170     1209  1
;  171     1210  1
;  172     1211  1 MACRO
;  173     1212  1
;  174     1213  1         ! Structure of TYPE parameter to TREE routine.
;  175     1214  1         !
;  176     1215  1         B_TYPE=         0,0,8,0 %,      ! Expected record type
;  177     1216  1         V_NO_MARK=      0,8,1,0 %,      ! Inhibit entering in record type map
;  178     1217  1         V_NO_QUEUE=     0,9,1,0 %,      ! Inhibit enqueuing jobs
;  179     1218  1         V_NO_TRAVERSE=  0,10,1,0 %,     ! Inhibit traversing list
;  180     1219  1         V_SMQ_P2=       0,11,1,0 %,     ! SMQ pass 2
```

```
;  181          1220  1          V_SMQ_P3=       0,12,1,0 %;     ! SMQ pass 3
;  182          1221  1
;  183          1222  1
;  184          1223  1 LITERAL
;  185          1224  1          M_NO_MARK=      $FIELDMASK(V_NO_MARK),
;  186          1225  1          M_NO_QUEUE=     $FIELDMASK(V_NO_QUEUE),
;  187          1226  1          M_NO_TRAVERSE=  $FIELDMASK(V_NO_TRAVERSE),
;  188          1227  1          M_SMQ_P2=       $FIELDMASK(V_SMQ_P2),
;  189          1228  1          M_SMQ_P3=       $FIELDMASK(V_SMQ_P3);
;  190          1229  1
;  191          1230  1
;  192          1231  1 BUILTIN
;  193          1232  1          EDIV,
;  194          1233  1          TESTBITSC,
;  195          1234  1          TESTBITSS;
```

```
;  197     1235  1 ROUTINE REFERENCE_ENTRY_NUMBER(ENTRY_NUMBER): L_TREE NOVALUE=
;  198     1236  1
;  199     1237  1 !++
;  200     1238  1 !
;  201     1239  1 ! FUNCTIONAL DESCRIPTION:
;  202     1240  1 !       This routine notes a reference to an entry number during the cold start
;  203     1241  1 !       of an existing file.
;  204     1242  1 !
;  205     1243  1 ! INPUT PARAMETERS:
;  206     1244  1 !       ENTRY_NUMBER    - Entry number.
;  207     1245  1 !
;  208     1246  1 ! IMPLICIT INPUTS:
;  209     1247  1 !       SQH             - Pointer to SQH.
;  210     1248  1 !
;  211     1249  1 ! OUTPUT PARAMETERS:
;  212     1250  1 !       NONE
;  213     1251  1 !
;  214     1252  1 ! IMPLICIT OUTPUTS:
;  215     1253  1 !       NONE
;  216     1254  1 !
;  217     1255  1 ! ROUTINE VALUE:
;  218     1256  1 !       NONE
;  219     1257  1 !
;  220     1258  1 ! SIDE EFFECTS:
;  221     1259  1 !       NONE
;  222     1260  1 !
;  223     1261  1 !--
;  224     1262  1
;  225     1263  2 BEGIN
;  226     1264  2 EXTERNAL REGISTER
;  227     1265  2         SQH = 11:       REF BBLOCK;     ! Pointer to SQH
;  228     1266  2 LOCAL
;  229     1267  2         BIT_NUMBER;                     ! Normalized entry number
;  230     1268  2
;  231     1269  2
;  232     1270  2 ! Range check the entry number.
;  233     1271  2 !
;  234     1272  2 IF .ENTRY_NUMBER EQLU 0
;  235     1273  2 OR .ENTRY_NUMBER GTRU .SQH[SQH$L_HIGHEST_ENTRY_NUMBER]
;  236     1274  2 THEN
;  237     1275  2     RETURN;
;  238     1276  2
;  239     1277  2
;  240     1278  2 ! Normalize the entry number to a bit number.  If it lies in the queue
;  241     1279  2 ! header, set it there.  Otherwise, convert it to an extension bitmap
;  242     1280  2 ! block offset and bit number and set it there.
;  243     1281  2 !
;  244     1282  2 BIT_NUMBER = .ENTRY_NUMBER - 1;
;  245     1283  2 IF .BIT_NUMBER LSSU SQH$S_ENTRY_BITMAP * 8
;  246     1284  2 THEN
;  247     1285  2     BITVECTOR[SQH[SQH$B_ENTRY_BITMAP], .BIT_NUMBER] = TRUE
;  248     1286  2 ELSE
;  249     1287  3     BEGIN
;  250     1288  3     LOCAL
;  251     1289  3         BLOCK_NUMBER,               ! Offset in bitmap vector
;  252     1290  3         Q:              VECTOR[2],  ! Temporary for EDIV
;  253     1291  3         SEB_N,                      ! Record number of extension bitmap
```

```
;  254          1292  3          SEB:              REF BBLOCK;     ! Pointer to extension bitmap
;  255          1293  3
;  256          1294  3
;  257          1295  3      Q[0] = .BIT_NUMBER - SQH$S_ENTRY_BITMAP * 8;
;  258          1296  3      Q[1] = 0;
;  259          1297  3      EDIV(%REF(SYM$S_DATA * 8), Q, BLOCK_NUMBER, BIT_NUMBER);
;  260          1298  3      IF .BLOCK_NUMBER LSSU SQH$S_ENTRY_BITMAP_VECTOR74
;  261          1299  3      THEN
;  262          1300  4          BEGIN
;  263          1301  4          SEB_N = .VECTOR[SQH[SQH$L_ENTRY_BITMAP_VECTOR], .BLOCK_NUMBER];
;  264          1302  4          IF .SEB_N NEQ 0
;  265          1303  4          THEN
;  266          1304  5              BEGIN
;  267          1305  5              SEB = READ_RECORD(.SEB_N);
;  268          1306  5              BITVECTOR[SEB[SYM$T_DATA], .BIT_NUMBER] = TRUE;
;  269          1307  5              REWRITE_RECORD(.SEB_N);
;  270          1308  4              END;
;  271          1309  3          END;
;  272          1310  2      END;
;  273          1311  1 END;


                                                .TITLE  INITQUEUE Initialize system job queue file
                                                .IDENT  \V04-001\

                                                .PSECT  COMMON,NOEXE,  OVR,2

                                        00000 DIAG_STORAGE_BASE:
                                                .BLKB   0
                                        00000 DIAG_TRACE:
                                                .BLKB   96
                                        00060 DIAG_COUNT:
                                                .BLKB   96
                                        000C0 DIAG_FLAGS:
                                                .BLKB   4
                                        000C4 WORK_AREA:
                                                .BLKB   44
                                        000F0 SNDJBC_COUNT:
                                                .BLKB   132
                                        00174 GETQUI_COUNT:
                                                .BLKB   40
                                        0019C SNDACC_COUNT:
                                                .BLKB   28
                                        001B8 SNDSMB_COUNT:
                                                .BLKB   72
                                        00200 DIAG_STORAGE_END:
                                                .BLKB   0
                                        00200 FLAGS:    .BLKB   4
                                        00204 IMAGE_DUMP_STSFLG:
                                                .BLKB   4
                                        00208 THIS_SYSID:
                                                .BLKB   6
                                        0020E           .BLKB   2
                                        00210 CUR_TIME:
                                                .BLKB   8
                                        00218 HOURLY_TIME:
                                                .BLKB   8
```

```
00220 HOURLY_PARAMS:
                .BLKB   20
00234 SYMBIONT_COUNT:
                .BLKB   4
00238 QUEUE_REFERENCE_COUNT:
                .BLKB   4
0023C MBX_MESSAGE_COUNT:
                .BLKB   4
00240 MBX:      .BLKB   4
00244 MBX_END:  .BLKB   4
00248 MEMORY_FREE_QUEUES:
                .BLKB   40
00270 NONAST_WORK_QUEUE:
                .BLKB   8
00278 BCB_FREE_LIST:
                .BLKB   4
0027C BCB_ACTIVE_LIST:
                .BLKB   4
00280 GQL_FREE_LIST:
                .BLKB   4
00284 GQL_ACTIVE_LIST:
                .BLKB   4
00288 OPEN_GETQUI_LIST:
                .BLKB   4
0028C PROCESS_DATA_LIST:
                .BLKB   4
00290 SYMBIONT_CONTROL:
                .BLKB   4
00294 SPARE_AREA:
                .BLKB   12
002A0 REMOTE_REQUEST_LKSB:
                .BLKB   8
002A8 QUEUE_FILE_LKSB:
                .BLKB   8
002B0 QUEUE_LOCK_LKSB:
                .BLKB   8
002B8 RSP:      .BLKB   8
002C0 JBC_PRIORITY:
                .BLKB   4
002C4 JBC_PRIVILEGES:
                .BLKB   8
002CC JBC_QUOTAS:
                .BLKB   66
0030E           .BLKB   2
00310 JBC_UIC:  .BLKB   4
00314 QUEUE_FAB:
                .BLKB   80
00364 QUEUE_RAB:
                .BLKB   68
003A8 QUEUE_NAM:
                .BLKB   96
00408 QUEUE_XAB:
                .BLKB   88
00460 QUEUE_RSA:
                .BLKB   255
0055F           .BLKB   1
00560 QUEUE_ALQ:
```

```
                        .BLKB   4
00564 QUEUE_MBF:
                        .BLKB   1
00565                   .BLKB   3
00568 ACCOUNTING_FABS:
                        .BLKB   8
00570 ACCOUNTING_RABS:
                        .BLKB   8
00578 ACCOUNT_FAB_A:
                        .BLKB   80
005C8 ACCOUNT_RAB_A:
                        .BLKB   68
0060C ACCOUNT_NAM_A:
                        .BLKB   96
0066C ACCOUNT_RSA_A:
                        .BLKB   255
0076B                   .BLKB   1
0076C ACCOUNT_FAB_B:
                        .BLKB   80
007BC ACCOUNT_RAB_B:
                        .BLKB   68
00800 ACCOUNT_NAM_B:
                        .BLKB   96
00860 ACCOUNT_RSA_B:
                        .BLKB   255
0095F                   .BLKB   1
00960 DIAG_FAB:
                        .BLKB   80
009B0 DIAG_RAB:
                        .BLKB   68
009F4 MBX_CHAN:
                        .BLKB   4
009F8 MBX_IOSB:
                        .BLKB   8
00A00 MBX_BUFFER:
                        .BLKB   1024
00E00 VALUE_STORAGE_BASE:
                        .BLKB   0
00E00 ITEM_PRESENT:
                        .BLKB   32
00E20 VALUE_GETQUI_BASE:
                        .BLKB   0
00E20 VALUE_ACCOUNTING_MESSAGE:
                        .BLKB   6
00E26 VALUE_ACCOUNTING_TYPES:
                        .BLKB   4
00E2A VALUE_AFTER_TIME:
                        .BLKB   8
00E32 VALUE_ALIGNMENT_PAGES:
                        .BLKB   1
00E33 VALUE_BASE_PRIORITY:
                        .BLKB   1
00E34 VALUE_BATCH_INPUT:
                        .BLKB   6
00E3A VALUE_BATCH_OUTPUT:
                        .BLKB   10
00E44 VALUE_BUFFER_COUNT:
```

```
                        .BLKB   1
00E45 VALUE_CHARACTERISTIC_NAME:
                        .BLKB   6
00E4B VALUE_CHARACTERISTIC_NUMBER:
                        .BLKB   1
00E4C VALUE_CHARACTERISTICS:
                        .BLKB   16
00E5C VALUE_CHECKPOINT_DATA:
                        .BLKB   8
00E62 VALUE_CLI:
                        .BLKB   6
00E68 VALUE_CPU_DEFAULT:
                        .BLKB   4
00E6C VALUE_CPU_LIMIT:
                        .BLKB   4
00E70 VALUE_DESTINATION_QUEUE:
                        .BLKB   8
00E78 VALUE_DEVICE_NAME:
                        .BLKB   6
00E7E VALUE_ENTRY_NUMBER:
                        .BLKB   4
00E82 VALUE_ENTRY_NUMBER_OUTPUT:
                        .BLKB   10
00E8C VALUE_EXTEND_QUANTITY:
                        .BLKB   2
00E8E VALUE_FILE_COPIES:
                        .BLKB   1
00E8F VALUE_FILE_IDENTIFICATION:
                        .BLKB   36
00EB3 VALUE_FILE_SETUP_MODULES:
                        .BLKB   6
00EB9 VALUE_FILE_SPECIFICATION:
                        .BLKB   6
00EBF VALUE_FIRST_PAGE:
                        .BLKB   4
00EC3 VALUE_FORM_DESCRIPTION:
                        .BLKB   6
00EC9 VALUE_FORM_LENGTH:
                        .BLKB   1
00ECA VALUE_FORM_MARGIN_BOTTOM:
                        .BLKB   1
00ECB VALUE_FORM_MARGIN_LEFT:
                        .BLKB   2
00ECD VALUE_FORM_MARGIN_RIGHT:
                        .BLKB   2
00ECF VALUE_FORM_MARGIN_TOP:
                        .BLKB   1
00ED0 VALUE_FORM_NAME:
                        .BLKB   6
00ED6 VALUE_FORM_NUMBER:
                        .BLKB   4
00EDA VALUE_FORM:
                        .BLKB   8
00EE2 VALUE_FORM_SETUP_MODULES:
                        .BLKB   6
00EE8 VALUE_FORM_STOCK:
                        .BLKB   6
```

```
00EEE VALUE_FORM_WIDTH:
                .BLKB   2
00EF0 VALUE_GENERIC_TARGET:
                .BLKB   996
012D4 VALUE_JOB_COPIES:
                .BLKB   1
012D5 VALUE_JOB_LIMIT:
                .BLKB   1
012D6 VALUE_JOB_NAME:
                .BLKB   6
012DC VALUE_JOB_RESET_MODULES:
                .BLKB   6
012E2 VALUE_JOB_SIZE_MAXIMUM:
                .BLKB   4
012E6 VALUE_JOB_SIZE_MINIMUM:
                .BLKB   4
012EA VALUE_JOB_STATUS_OUTPUT:
                .BLKB   TO
01cF4 VALUE_LAST_PAGE:
                .BLKB   4
012F8 VALUE_LIBRARY_SPECIFICATION:
                .BLKB   6
012FE VALUE_LOG_QUEUE:
                .BLKB   8
01306 VALUE_LOG_SPECIFICATION:
                .BLKB   6
0130C VALUE_NOTE:
                .BLKB   6
01312 VALUE_OPERATOR_REQUEST:
                .BLKB   6
01318 VALUE_OWNER_UIC:
                .BLKB   4
0131C VALUE_PAGE_SETUP_MODULES:
                .BLKB   6
01322 VALUE_PARAMETER_1:
                .BLKB   6
01328 VALUE_PARAMETER_2:
                .BLKB   6
0132E VALUE_PARAMETER_3:
                .BLKB   6
01334 VALUE_PARAMETER_4:
                .BLKB   6
0133A VALUE_PARAMETER_5:
                .BLKB   6
01340 VALUE_PARAMETER_6:
                .BLKB   6
01346 VALUE_PARAMETER_7:
                .BLKB   6
0134C VALUE_PARAMETER_8:
                .BLKB   6
01352 VALUE_PRIORITY:
                .BLKB   1
01353 VALUE_PROCESSOR:
                .BLKB   6
01359 VALUE_PROTECTION:
                .BLKB   4
0135D VALUE_QUEUE:
```

```
                        .BLKB   6
01363 VALUE_QUEUE_FILE_SPECIFICATION:
                        .BLKB   6
01369 VALUE_RELATIVE_PAGE:
                        .BLKB   4
0136D VALUE_RESERVED_INPUT_1:
                        .BLKB   1
0136E VALUE_RESERVED_INPUT_2:
                        .BLKB   2
01370 VALUE_RESERVED_INPUT_3:
                        .BLKB   4
01374 VALUE_RESERVED_INPUT_4:
                        .BLKB   6
0137A VALUE_RESERVED_OUTPUT_1:
                        .BLKB   10
01384 VALUE_RESERVED_OUTPUT_2:
                        .BLKB   10
0138E VALUE_SEARCH_STRING:
                        .BLKB   6
01394 VALUE_SCSNODE_NAME:
                        .BLKB   6
0139A VALUE_WSDEFAULT:
                        .BLKB   2
0139C VALUE_WSEXTENT:
                        .BLKB   2
0139E VALUE_WSQUOTA:
                        .BLKB   2
013A0 VALUE_STORAGE_END:
                        .BLKB   0

                        .PSECT  CODE,NOWRT,2

00000001  00000 P.AAA:  .LONG   1
00000002  00004 P.AAB:  .LONG   2
00000003  00008 P.AAC:  .LONG   3

                JBC$_CLOSEOUT=      266328
                JBC$_NOCMKRNL=      272388
                JBC$_NOOPER=        272532
                JBC$_NOSYSNAM=      272404
                JBC$_OPENIN=        266392
                JBC$_OPENOUT=       266400
                JBC$_READERR=       266416
                JBC$_WRITEERR=      266448
                SQH_KBF=            P.AAA
                SFM_KBF=            P.AAB
                SFX_KBF=            P.AAC
                        .EXTRN  AFTER_AST, COMPLETE_JOB
                        .EXTRN  DEALLOCATE_MEMORY
                        .EXTRN  DELETE_SJH_RECORD
                        .EXTRN  ENQUEUE_JOB, FLUSH_RECORD
                        .EXTRN  JOB_SCHEDULING_POLICY
                        .EXTRN  LOCK_QUEUE_FILE
                        .EXTRN  QUEUE_MASTER_AST
                        .EXTRN  READ_RECORD, RELEASE_RECORD
                        .EXTRN  REMOTE_BLOCKING_AST
                        .EXTRN  REWRITE_RECORD, SCAN_INCOMPLETE_SERVICES
```

```
                                                    .EXTRN  SIGNAL_FILE_ERROR
                                                    .EXTRN  UPDATE_GETQUI_DATA
                                                    .EXTRN  WRITE_ACCOUNTING_RECORD

                                 000C 00000 REFERENCE_ENTRY_NUMBER:
                                                    .WORD   Save R2,R3                               ; 1235
                5E          08  C2 00002            SUBL2   #8, SP                                   ;
                52     04   AC  D0 00005            MOVL    ENTRY_NUMBER, R2                         ; 1272
                            4C  13 00009            BEQL    3$                                       ;
          3C    AB          52  D1 0000B            CMPL    R2, 60(SQH)                              ; 1273
                            46  1A 0000F            BGTRU   3$                                       ;
                            52  D7 00011            DECL    BIT_NUMBER                               ; 1282
    00000800    8F          52  D1 00013            CMPL    BIT_NUMBER, #2048                        ; 1283
                            07  1E 0001A            BGEQU   1$                                       ;
  35    0100    CB          52  E2 00^1C            BBSS    BIT_NUMBER, 256(SQH), 3$                 ; 1285
                                04 00022            RET                                              ;
                6E   F800   C2  9E 00023 1$:        MOVAB   -2048(R2), Q                             ; 1295
                       04   AE  D4 00028            CLRL    Q+4                                      ; 1296
52    50        6E 00000FA0 8F  7B 0002B            EDIV    #4000, Q, BLOCK_NUMBER, BIT_NUMBER       ; 1297
                08          50  D1 00034            CMPL    BLOCK_NUMBER, #8                         ; 1298
                            1E  1E 00037            BGEQU   3$                                       ;
                53     14 AB40  D0 00039            MOVL    20(SQH)[BLOCK_NUMBER], SEB_N             ; 1301
                            17  13 0003E            BEQL    3$                                       ; 1302
                            53  DD 00040            PUSHL   SEB_N                                    ; 1305
    00000000G   EF          01  FB 00042            CALLS   #1, READ_RECORD                          ;
  00      0C    A0          52  E2 00049            BBSS    BIT_NUMBER, 12(SEB), 2$                  ; 1306
                            53  DD 0004E 2$:        PUSHL   SEB_N                                    ; 1307
    00000000G   EF          01  FB 00050            CALLS   #1, REWRITE_RECORD                       ;
                                04 00057 3$:        RET                                              ; 1311

; Routine Size:  88 bytes,    Routine Base:  CODE + 000C
```

```
;  275     1312  1 ROUTINE ENQUEUE(SMQ,SJH_N,SJH): L_TREE NOVALUE=
;  276     1313  1
;  277     1314  1 !++
;  278     1315  1 !
;  279     1316  1 ! FUNCTIONAL DESCRIPTION:
;  280     1317  1 !       This routine enqueues a job during the cold start of an existing file.
;  281     1318  1 !
;  282     1319  1 ! INPUT PARAMETERS:
;  283     1320  1 !       SMQ             - Pointer to job's SMQ.
;  284     1321  1 !       SJH_N           - Record number of SJH.
;  285     1322  1 !       SJH             - Pointer to SJH.
;  286     1323  1 !
;  287     1324  1 ! IMPLICIT INPUTS:
;  288     1325  1 !       SQH             - Pointer to SQH.
;  289     1326  1 !
;  290     1327  1 ! OUTPUT PARAMETERS:
;  291     1328  1 !       NONE
;  292     1329  1 !
;  293     1330  1 ! IMPLICIT OUTPUTS:
;  294     1331  1 !       NONE
;  295     1332  1 !
;  296     1333  1 ! ROUTINE VALUE:
;  297     1334  1 !       NONE
;  298     1335  1 !
;  299     1336  1 ! SIDE EFFECTS:
;  300     1337  1 !       NONE
;  301     1338  1 !
;  302     1339  1 !--
;  303     1340  1
;  304     1341  2 BEGIN
;  305     1342  2 MAP
;  306     1343  2         SMQ:            REF BBLOCK,     ! Pointer to SMQ
;  307     1344  2         SJH:            REF BBLOCK;     ! Pointer to SJH
;  308     1345  2 EXTERNAL REGISTER
;  309     1346  2         SQH = 11:       REF BBLOCK;     ! Pointer to SQH
;  310     1347  2
;  311     1348  2
;  312     1349  2 ! Reinitialize the forward link.
;  313     1350  2 !
;  314     1351  2 SJH[SYM$L_LINK] = 0;
;  315     1352  2
;  316     1353  2
;  317     1354  2 ! If appropriate, enter the job on the hold queue.
;  318     1355  2 !
;  319     1356  2 IF .SJH[SJH$V_HOLDING]
;  320     1357  2 OR .SJH[SJH$V_REFUSED]
;  321     1358  2 OR .SJH[SJH$V_RETAINED]
;  322     1359  2 THEN
;  323     1360  3     BEGIN
;  324     1361  3     LOCAL
;  325     1362  3         SJH_NP,                         ! Record number of predecessor of SJH
;  326     1363  3         SJH_P:          REF BBLOCK;     ! Pointer to predecessor of SJH
;  327     1364  3
;  328     1365  3
;  329     1366  3     IF .SMQ[SMQ$L_HOLD_LIST] EQL 0
;  330     1367  3     THEN
;  331     1368  3         SMQ[SMQ$L_HOLD_LIST] = .SJH_N
```

```
;   332          1369  3      ELSE
;   333          1370  4          BEGIN
;   334          1371  4          SJH_P = READ_RECORD(SJH_NP = .SMQ[SMQ$L_HOLD_LIST_END]);
;   335          1372  4          SJH_P[SYM$L_LINK] = .SJH_N;
;   336          1373  4          REWRITE_RECORD(.SJH_NP);
;   337          1374  3          END;
;   338          1375  3      SMQ[SMQ$L_HOLD_LIST_END] = .SJH_N;
;   339          1376  3      END
;   340          1377  3
;   341          1378  3
;   342          1379  3  ! Enter the job on the pending or timer queue, as appropriate.
;   343          1380  3  !
;   344          1381  2  ELSE
;   345          1382  3      BEGIN
;   346          1383  3      LOCAL
;   347          1384  3          TIMER_QUEUE,                        ! True if entering on timer queue
;   348          1385  3          LIST_HEAD:      REF VECTOR,         ! Pointer to appropriate list head
;   349          1386  3          SJH_NP,                             ! Record number of precedessor of SJH
;   350          1387  3          SJH_P:          REF BBLOCK,         ! Pointer to predecessor of SJH
;   351          1388  3          SJH_NS,                             ! Record number of successor of SJH
;   352          1389  3          SJH_S:          REF BBLOCK;         ! Pointer to successor of SJH
;   353          1390  3
;   354          1391  3
;   355          1392  3      ! Establish the queue on which the job should be entered and increase
;   356          1393  3      ! the reference count.
;   357          1394  3      !
;   358          1395  4      IF TIME_GTRU(SJH[SJH$Q_AFTER_TIME], CUR_TIME)
;   359          1396  3      THEN
;   360          1397  4          BEGIN
;   361          1398  4          TIMER_QUEUE = TRUE;
;   362          1399  4          LIST_HEAD = SQH[SQH$L_TIMER_LIST];
;   363          1400  4          SMQ[SMQ$W_TIMER_JOB_COUNT] = .SMQ[SMQ$W_TIMER_JOB_COUNT] + 1;
;   364          1401  4          END
;   365          1402  3      ELSE
;   366          1403  4          BEGIN
;   367          1404  4          TIMER_QUEUE = FALSE;
;   368          1405  4          IF .SMQ[SMQ$V_BATCH]
;   369          1406  4              THEN LIST_HEAD = SQH[SQH$L_PENDING_BATCH_LIST]
;   370          1407  4              ELSE LIST_HEAD = SQH[SQH$L_PENDING_PRINT_LIST];
;   371          1408  4          SMQ[SMQ$W_PENDING_JOB_COUNT] = .SMQ[SMQ$W_PENDING_JOB_COUNT] + 1;
;   372          1409  3          END;
;   373          1410  3
;   374          1411  3
;   375          1412  3      ! Check for the special case that the job goes at the end.
;   376          1413  3      !
;   377          1414  3      IF .LIST_HEAD[1] NEQ 0
;   378          1415  3      THEN
;   379          1416  4          BEGIN
;   380          1417  4          SJH_S = READ_RECORD(SJH_NS = .LIST_HEAD[1]);
;   381          1418  4          IF
;   382          1419  5              BEGIN
;   383          1420  5              IF .TIMER_QUEUE
;   384          1421  6                  THEN TIME_GEQU(SJH[SJH$Q_AFTER_TIME], SJH_S[SJH$Q_AFTER_TIME])
;   385          1422  5                  ELSE JOB_SCHEDULING_POLICY(.SMQ, .SJH_S, .SJH)
;   386          1423  5              END
;   387          1424  4          THEN
;   388          1425  5              BEGIN
```

```
;  389          1426  5              SJH_S[SYM$L_LINK] = .SJH_N;
;  390          1427  5              LIST_HEAD[1] = .SJH_N;
;  391          1428  5              REWRITE_RECORD(.SJH_NS);
;  392          1429  5              RETURN;
;  393          1430  4              END;
;  394          1431  3          END.
;  395          1432  3
;  396          1433  3
;  397          1434  3      ! Search for the correct insertion point.
;  398          1435  3      !
;  399          1436  3      SJH_NP = SQH$K_RECNO;
;  400          1437  3      SJH_P = .SQH;
;  401          1438  3      SJH_NS = .LIST_HEAD[0];
;  402          1439  3      WHILE .SJH_NS NEQ 0 DO
;  403          1440  4          BEGIN
;  404          1441  4          SJH_S = READ_RECORD(.SJH_NS);
;  405          1442  4          IF
;  406          1443  5              BEGIN
;  407          1444  5              IF .TIMER_QUEUE
;  408          1445  6                  THEN TIME_GTRU(SJH_S[SJH$Q_AFTER_TIME], SJH[SJH$Q_AFTER_TIME])
;  409          1446  5                  ELSE JOB_SCHEDULING_POLICY(.SMQ, .SJH, .SJH_S)
;  410          1447  5              END
;  411          1448  4          THEN
;  412          1449  5              BEGIN
;  413          1450  5              RELEASE_RECORD(.SJH_NS);
;  414          1451  5              EXITLOOP;
;  415          1452  4              END;
;  416          1453  4          IF .SJH_NP NEQ SQH$K_RECNO THEN RELEASE_RECORD(.SJH_NP);
;  417          1454  4          SJH_NP = .SJH_NS;
;  418          1455  4          SJH_P = .SJH_S;
;  419          1456  4          SJH_NS = .SJH_S[SYM$L_LINK];
;  420          1457  3          END;
;  421          1458  3
;  422          1459  3
;  423          1460  3      ! Enqueue the job at the correct insertion point.
;  424          1461  3      !
;  425          1462  3      IF .SJH_NP EQL SQH$K_RECNO
;  426          1463  3      THEN
;  427          1464  3          LIST_HEAD[0] = .SJH_N
;  428          1465  3      ELSE
;  429          1466  4          BEGIN
;  430          1467  4          SJH_P[SYM$L_LINK] = .SJH_N;
;  431          1468  4          REWRITE_RECORD(.SJH_NP);
;  432          1469  3          END;
;  433          1470  3      SJH[SYM$L_LINK] = .SJH_NS;
;  434          1471  3      IF .SJH_NS EQL 0 THEN LIST_HEAD[1] = .SJH_N;
;  435          1472  2      END;
;  436          1473  1 END;


                                  07FC 00000 ENQUEUE:.WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10          ; 1312
                        55      0C   AC  D0 00002          MOVL    SJH, R5                            ; 1351
                                     65  D4 00006          CLRL    (R5)                               ;
                        52      04   AC  D0 00008          MOVL    SMQ, R2                            ; 1366
```

```
                    59      08    AC  D0 0000C          MOVL    SJH_N, R9                       ; 1368
              0A    10      A5    05  E0 00010          BBS     #5, 16(R5), 1$                  ; 1356
                            10    A5  95 00015          TSTB    16(R5)                          ; 1357
                                  05  19 00018          BLSS    1$                              ;
              29    11      A5    03  E1 0001A          BBC     #3, 17(R5), 4$                  ; 1358
                            78    A2  D5 0001F 1$:      TSTL    120(R2)                         ; 1366
                                  06  12 00022          BNEQ    2$                              ;
                    78      A2    59  D0 00024          MOVL    R9, 120(R2)                     ; 1368
                                  19  11 00028          BRB     3$                              ;
                    53      7C    A2  D0 0002A 2$:      MOVL    124(R2), SJH_NP                 ; 1371
                                  53  DD 0002E          PUSHL   SJH_NP                          ;
          00000000G   EF            01  FB 00030          CALLS   #1, READ_RECORD                 ;
                      60            59  D0 00037          MOVL    R9, (SJH_P)                     ; 1372
                                  53  DD 0003A          PUSHL   SJH_NP                          ; 1373
          00000000G   EF            01  FB 0003C          CALLS   #1, REWRITE_RECORD              ;
                    7C      A2    59  D0 00043 3$:      MOVL    R9, 124(R2)                     ; 1375
                                  04  00047          RET                                     ; 1356
                    57      009C  C5  D0 00048 4$:      MOVL    156(R5), R7                     ; 1395
          00000000'   EF            57  D1 0004D          CMPL    R7, CUR_TIME+4                  ;
                                  0D  1A 00054          BGTRU   5$                              ;
                                  18  12 00056          BNEQ    6$                              ;
          00000000'   EF    0098    C5  D1 00058          CMPL    152(R5), CUR_TIME               ;
                                  0D  1B 00061          BLEQU   6$                              ;
                      5A            01  D0 00063 5$:      MOVL    #1, TIMER_QUEUE                 ; 1398
                      54      68    AB  9E 00066          MOVAB   104(R11), LIST_HEAD             ; 1399
                            010C    C2  B6 0006A          INCW    268(R2)                         ; 1400
                                  14  11 0006E          BRB     9$                              ; 1395
                                  5A  D4 00070 6$:      CLRL    TIMER_QUEUE                     ; 1404
                      06      0C    A2  E9 00072          BLBC    12(R2), 7$                      ; 1405
                      54      54    AB  9E 00076          MOVAB   84(R11), LIST_HEAD              ; 1406
                                  04  11 0007A          BRB     8$                              ;
                      54      5C    AB  9E 0007C 7$:      MOVAB   92(R11), LIST_HEAD              ; 1407
                            0102    C2  B6 00080 8$:      INCW    258(R2)                         ; 1408
                              04    A4  D5 00084 9$:      TSTL    4(LIST_HEAD)                    ; 1414
                                  46  13 00087          BEQL    12$                             ;
                    56        04    A4  D0 00089          MOVL    4(LIST_HEAD), SJH_NS            ; 1417
                                  56  DD 0008D          PUSHL   SJH_NS                          ;
      00000000G   EF                01  FB 0008F          CALLS   #1, READ_RECORD                 ;
                  53                50  D0 00096          MOVL    R0, SJH_S                       ;
                  14                5A  E9 00099          BLBC    TIMER_QUEUE, 10$                ; 1420
            009C  C3                57  D1 0009C          CMPL    R7, 156(SJH_S)                  ; 1421
                                  19  1A 000A1          BGTRU   11$                             ;
                                  2A  12 000A3          BNEQ    12$                             ;
            0098  C3    0098      C5  D1 000A5          CMPL    152(R5), 152(SJH_S)             ;
                                  21  1F 000AC          BLSSU   12$                             ;
                                  0C  11 000AE          BRB     11$                             ;
                                  2C  BB 000B0 10$:     PUSHR   #^M<R2,R3,R5>                   ; 1422
      00000000G   EF                03  FB 000B2          CALLS   #3, JOB_SCHEDULING_POLICY       ;
                  13                50  E9 000B9          BLBC    R0, 12$                         ;
                  63      08        AC  D0 000BC 11$:     MOVL    SJH_N, (SJH_S)                  ; 1426
              04  A4      08        AC  D0 000C0          MOVL    SJH_N, 4(LIST_HEAD)             ; 1427
                                  56  DD 000C5          PUSHL   SJH_NS                          ; 1428
      00000000G   EF                01  FB 000C7          CALLS   #1, REWRITE_RECORD              ;
                                  04  000CE          RET                                     ; 1425
                  57                01  D0 000CF 12$:     MOVL    #1, SJH_NP                      ; 1436
                  58                5B  D0 000D2          MOVL    SQH, SJH_P                      ; 1437
                  56                64  D0 000D5          MOVL    (LIST_HEAD), SJH_NS             ; 1438
```

```
                                                 57  13 000D8 13$:     BEQL    18$                                     ; 1439
                                                 56  DD 000DA         PUSHL   SJH_NS                                  ; 1441
                    00000000G   EF               01  FB 000DC         CALLS   #1, READ_RECORD                         ;
                                53               50  D0 000E3         MOVL    R0, SJH_S                               ;
                                16               5A  E9 000E6         BLBC    TIMER_QUEUE, 14$                        ; 1445
                        009C    C5      009C     C3  D1 000E9         CMPL    156(SJH_S), 156(R5)                     ;
                                                 1B  1A 000F0         BGTRU   15$                                     ;
                                                 24  12 000F2         BNEQ    16$                                     ;
                        0098    C5      0098     C3  D1 000F4         CMPL    152(SJH_S), 152(R5)                     ;
                                                 1B  1B 000FB         BLEQU   16$                                     ;
                                                 0E  11 000FD         BRB     15$                                     ;
                                                 53  DD 000FF 14$:    PUSHL   SJH_S                                   ; 1446
                                                 24  BB 00101         PUSHR   #^M<R2,R5>                              ;
                    00000000G   EF               03  FB 00103         CALLS   #3, JOB_SCHEDULING_POLICY               ;
                                0B               50  E9 0010A         BLBC    R0, 16$                                 ;
                                                 56  DD 0010D 15$:    PUSHL   SJH_NS                                  ; 1450
                    00000000G   EF               01  FB 0010F         CALLS   #1, RELEASE_RECORD                      ;
                                                 19  11 00116         BRB     18$                                     ; 1449
                                01               57  D1 00118 16$:    CMPL    SJH_NP, #1                              ; 1453
                                                 09  13 0011B         BEQL    17$                                     ;
                                                 57  DD 0011D         PUSHL   SJH_NP                                  ;
                    00000000G   EF               01  FB 0011F         CALLS   #1, RELEASE_RECORD                      ;
                                57               56  D0 00126 17$:    MOVL    SJH_NS, SJH_NP                          ; 1454
                                58               53  D0 00129         MOVL    SJH_S, SJH_P                            ; 1455
                                56               63  D0 0012C         MOVL    (SJH_S), SJH_NS                         ; 1456
                                                 A7  11 0012F         BRB     13$                                     ; 1439
                                01               57  D1 00131 18$:    CMPL    SJH_NP, #1                              ; 1462
                                                 05  12 00134         BNEQ    19$                                     ;
                                64               59  D0 00136         MOVL    R9, (LIST_HEAD)                         ; 1464
                                                 0C  11 00139         BRB     20$                                     ;
                                68               59  D0 0013B 19$:    MOVL    R9, (SJH_P)                             ; 1467
                                                 57  DD 0013E         PUSHL   SJH_NP                                  ; 1468
                    00000000G   EF               01  FB 00140         CALLS   #1, REWRITE_RECORD                      ;
                                65               56  D0 00147 20$:    MOVL    SJH_NS, (R5)                            ; 1470
                                                 04  12 0014A         BNEQ    21$                                     ; 1471
                        04      A4               59  D0 0014C         MOVL    R9, 4(LIST_HEAD)                        ;
                                                     04 00150 21$:    RET                                             ; 1473

; Routine Size:  337 bytes,    Routine Base:  CODE + 0064
```

```
;  438          1474  1 ROUTINE TREE(RECNO,TYPE,DEL,P1): L_TREE NOVALUE=
;  439          1475  1
;  440          1476  1 !++
;  441          1477  1 !
;  442          1478  1 ! FUNCTIONAL DESCRIPTION:
;  443          1479  1 !       This routine recursively processes a record tree during the cold start
;  444          1480  1 !       of an existing file.
;  445          1481  1 !
;  446          1482  1 ! INPUT PARAMETERS:
;  447          1483  1 !       RECNO           - Record number.
;  448          1484  1 !       TYPE            - Expected record type and control flags.
;  449          1485  1 !       DEL             - True if this subtree is being deleted.
;  450          1486  1 !       P1              - Function-dependent parameter.
;  451          1487  1 !
;  452          1488  1 ! IMPLICIT INPUTS:
;  453          1489  1 !       SQH             - Pointer to SQH.
;  454          1490  1 !       RTM             - Pointer to record type map.
;  455          1491  1 !
;  456          1492  1 ! OUTPUT PARAMETERS:
;  457          1493  1 !       NONE
;  458          1494  1 !
;  459          1495  1 ! IMPLICIT OUTPUTS:
;  460          1496  1 !       NONE
;  461          1497  1 !
;  462          1498  1 ! ROUTINE VALUE:
;  463          1499  1 !       NONE
;  464          1500  1 !
;  465          1501  1 ! SIDE EFFECTS:
;  466          1502  1 !       NONE
;  467          1503  1 !
;  468          1504  1 !--
;  469          1505  1
;  470          1506  2 BEGIN
;  471          1507  2 MAP
;  472          1508  2         TYPE:           BBLOCK;         ! Control flags
;  473          1509  2 EXTERNAL REGISTER
;  474          1510  2         RTM = 10:       REF VECTOR[,BYTE,SIGNED], ! Record type map
;  475          1511  2         SQH = 11:       REF BBLOCK;     ! Pointer to SQH
;  476          1512  2 LOCAL
;  477          1513  2         REC_N;                          ! Current record number
;  478          1514  2
;  479          1515  2
;  480          1516  2 MACRO
;  481          1517  2
;  482          1518  2         ! Process extension records linked to a variable data area.
;  483          1519  2         !
;  484        M 1520  2         MARK_VARIABLE_DATA(LENGTH,ADDRESS,TYPE,DELETE)=
;  485        M 1521  2             BEGIN
;  486        M 1522  2             IF .BBLOCK[ADDRESS, FVDF_LENGTH] GTRU (LENGTH - 2)
;  487        M 1523  2             THEN
;  488        M 1524  2                 TREE(
;  489        M 1525  2                     .BBLOCK[ADDRESS, FVDF_LINK],
;  490        M 1526  2                     TYPE,
;  491        M 1527  2                     DELETE);
;  492          1528  2             END %,
;  493          1529  2
;  494          1530  2
```

```
;  495          1531  2          ! Check that a link to another record is either zero or valid.
;  496          1532  2          !
;  497        M 1533  2          VALIDATE_OPTIONAL_LINK(RECNO,TYPE)=
;  498        M 1534  2              BEGIN
;  499        M 1535  2              IF (RECNO) NEQ 0
;  500        M 1536  2              THEN
;  501        M 1537  2                  VALIDATE_REQUIRED_LINK(RECNO,TYPE);
;  502          1538  2              END %,
;  503          1539  2
;  504          1540  2
;  505          1541  2          ! Check that a link to another record contains a valid record number
;  506          1542  2          ! and that the record is of the correct type.
;  507          1543  2          !
;  508        M 1544  2          VALIDATE_REQUIRED_LINK(RECNO,TYPE)=
;  509        M 1545  2              BEGIN
;  510        M 1546  2              IF
;  511        M 1547  2                  BEGIN
;  512        M 1548  2                  IF (RECNO) LSSU 2
;  513        M 1549  2                  OR (RECNO) GTRU .SQH[SQH$L_HIGHEST_RECORD]
;  514        M 1550  2                  THEN
;  515        M 1551  2                      TRUE
;  516        M 1552  2                  ELSE
;  517        M 1553  2                      .RTM[(RECNO)-1] NEQ (TYPE)
;  518        M 1554  2                  END
;  519        M 1555  2              THEN
;  520        M 1556  2                  SIGNAL(JBC$_QUEFORMAT);
;  521          1557  2              END %;
;  522          1558  2
;  523          1559  2
;  524          1560  2 ! Loop over all records in the list.
;  525          1561  2 !
;  526          1562  2 REC_N = .RECNO;
;  527          1563  2 WHILE .REC_N NEQ 0 DO
;  528          1564  3     BEGIN
;  529          1565  3     LOCAL
;  530          1566  3         REC:              REF BBLOCK,     ! Pointer to current record
;  531          1567  3         MODIFIED,                         ! True if record modified
;  532          1568  3         DELETE,                           ! True if record should be deleted
;  533          1569  3         REC_NS;                           ! Record number of successor
;  534          1570  3
;  535          1571  3
;  536          1572  3     DIAG_TRACE[9] = .REC_N;               ! ***** diagnostic info *****
;  537          1573  3     ! Range check the record number.
;  538          1574  3     !
;  539          1575  3     IF .REC_N LSSU 2
;  540          1576  3     OR .REC_N GTRU .SQH[SQH$L_HIGHEST_RECORD]
;  541          1577  3     THEN
;  542          1578  3         SIGNAL(JBC$_QUEFORMAT);
;  543          1579  3
;  544          1580  3
;  545          1581  3     ! Read the record and initialize to process it.
;  546          1582  3     !
;  547          1583  3     REC = READ_RECORD(.REC_N);
;  548          1584  3     MODIFIED = FALSE;
;  549          1585  3     DELETE = .DEL;
;  550          1586  3     REC_NS = .REC[SYM$L_LINK];
;  551          1587  3
```

```
;  552          1588  3
;  553          1589  3      ! Check the record type.
;  554          1590  3      !
;  555          1591  3      IF .REC[SYM$B_TYPE] NEQ .TYPE[B_TYPE]
;  556          1592  3      THEN
;  557          1593  3          SIGNAL(JBC$_QUEFORMAT);
;  558          1594  3
;  559          1595  3
;  560          1596  3      ! Unless this action is inhibited, note the record type in the record type
;  561          1597  3      ! map and check for multiply allocated records.
;  562          1598  3      !
;  563          1599  3      IF NOT .TYPE[V_NO_MARK]
;  564          1600  3      THEN
;  565          1601  4          BEGIN
;  566          1602  4          IF .RTM[.REC_N-1] GEQ 0 THEN SIGNAL(JBC$_QUEFORMAT);
;  567          1603  4          RTM[.REC_N-1] = .REC[SYM$B_TYPE];
;  568          1604  3          END;
;  569          1605  3
;  570          1606  3
;  571          1607  3      ! Dispatch to type-dependent processing.
;  572          1608  3      !
;  573          1609  3      CASE .REC[SYM$B_TYPE] FROM SYM$K_SCX TO SYM$K_REFUSAL_REASON OF
;  574          1610  3          SET
;  575          1611  3
;  576          1612  3
;  577          1613  3          [INRANGE, OUTRANGE]:
;  578          1614  3              0;
;  579          1615  3
;  580          1616  3
;  581          1617  3          [SYM$K_SFM]:
;  582          1618  4              BEGIN
;  583          1619  4
;  584          1620  4              ! Traverse substructures of the form record.
;  585          1621  4              !
;  586        P 1622  4              MARK_VARIABLE_DATA(
;  587        P 1623  4                  SFM$S_FORM_SETUP_MODULES, REC[SFM$T_FORM_SETUP_MODULES],
;  588          1624  4                  SYM$K_FORM_SETUP_MODULES, FALSE);
;  589        P 1625  4              MARK_VARIABLE_DATA(
;  590        P 1626  4                  SFM$S_PAGE_SETUP_MODULES, REC[SFM$T_PAGE_SETUP_MODULES],
;  591          1627  4                  SYM$K_PAGE_SETUP_MODULES, FALSE);
;  592          1628  3              END;
;  593          1629  3
;  594          1630  3
;  595          1631  3          [SYM$K_SFX]:
;  596          1632  4              BEGIN
;  597          1633  4              LOCAL
;  598          1634  4                  SFE:            REF BBLOCK;     ! Pointer to SFE
;  599          1635  4
;  600          1636  4              SFE = REC[SYM$T_DATA];
;  601          1637  4              INCR SFE_N FROM 0 TO SFX$K_ENTRIES-1 DO
;  602          1638  5                  BEGIN
;  603          1639  5                  IF CH$RCHAR(SFE[SFX$T_NAME]) EQL 0
;  604          1640  5                  THEN
;  605          1641  5                      EXITLC
;  606          1642  5                  ELSE
;  607          1643  6                      BEGIN
;  608          1644  6                      TREE(.SFE[SFX$L_FORM_LINK], SYM$K_SFM, FALSE);
```

```
;  609          1645  6                    SFE = .SFE + SFX$S_SFX;
;  610          1646  5                    END;
;  611          1647  4                END;
;  612          1648  3            END;
;  613          1649  3
;  614          1650  3
;  615          1651  3        [SYM$K_SQX]:
;  616          1652  4            BEGIN
;  617          1653  4            LOCAL
;  618          1654  4                T:              BBLOCK[4],      ! Local type
;  619          1655  4                SQE:            REF BBLOCK;     ! Pointer to SQE
;  620          1656  4
;  621          1657  4            T = .TYPE;  T[B_TYPE] = SYM$K_SMQ;
;  622          1658  4            SQE = REC[SYM$T_DATA];
;  623          1659  4            INCR SQE_N FROM 0 TO SQX$K_ENTRIES-1 DO
;  624          1660  5                BEGIN
;  625          1661  5                IF CH$RCHAR(SQE[SQX$T_NAME]) EQL 0
;  626          1662  5                THEN
;  627          1663  5                    EXITLOOP
;  628          1664  5                ELSE
;  629          1665  6                    BEGIN
;  630          1666  6                    TREE(.SQE[SQX$L_QUEUE_LINK], .T, FALSE);
;  631          1667  6                    SQE = .SQE + SQX$S_SQX;
;  632          1668  5                    END;
;  633          1669  4                END;
;  634          1670  3            END;
;  635          1671  3
;  636          1672  3
;  637          1673  3        [SYM$K_SMQ]:
;  638          1674  4            BEGIN
;  639          1675  4
;  640          1676  4            ! Third queue header pass.  This pass traverses the current
;  641          1677  4            ! list, which must occur on a pass after the hold list because
;  642          1678  4            ! failed jobs that are retained are enqueued to the hold list.
;  643          1679  4            !
;  644          1680  4            IF .TYPE[V_SMQ_P3]
;  645          1681  4            THEN
;  646          1682  5                BEGIN
;  647          1683  5                TREE(.REC[SMQ$L_CURRENT_LIST], SYM$K_SJH, FALSE);
;  648          1684  5                REC[SMQ$L_CURRENT_LIST] = 0;
;  649          1685  5                REC[SMQ$L_CURRENT_LIST_END] = 0;
;  650          1686  5                END
;  651          1687  5
;  652          1688  5
;  653          1689  5            ! Second queue header pass.  This pass traverses all substructures
;  654          1690  5            ! except the current list.
;  655          1691  5            !
;  656          1692  4            ELSE IF .TYPE[V_SMQ_P2]
;  657          1693  4            THEN
;  658          1694  5                BEGIN
;  659        P 1695  5                VALIDATE_OPTIONAL_LINK(
;  660          1696  5                    .REC[SMQ$L_ASSIGNED_QUEUE_LINK], SYM$K_SMQ);
;  661        P 1697  5                VALIDATE_OPTIONAL_LINK(
;  662          1698  5                    .REC[SMQ$L_FORM_LINK], SYM$K_SFM);
;  663          1699  5                TREE(.REC[SMQ$L_GENERIC_TARGET], SYM$K_GENERIC_TARGET, FALSE);
;  664          1700  5                TREE(.REC[SMQ$L_HOLD_LIST], SYM$K_SJH OR M_NO_QUEUE, FALSE);
;  665        P 1701  5                MARK_VARIABLE_DATA(
```

```
;  666        P 1702  5                    SMQ$S_JOB_RESET_MODULES, REC[SMQ$T_JOB_RESET_MODULES],
;  667          1703  5                    SYM$K_JOB_RESET_MODULES, FALSE);
;  668          1704  5                END
;  669          1705  5
;  670          1706  5
;  671          1707  5            ! First queue header pass.  This reinitializes fields and notes
;  672          1708  5            ! the record number.
;  673          1709  5            !
;  674          1710  4            ELSE
;  675          1711  5                BEGIN
;  676          1712  5                REC[SMQ$L_STATUS] = 0;
;  677          1713  5                REC[SMQ$V_STOPPED] = TRUE;
;  678          1714  5                CLEAR_TIME(REC[SMQ$Q_ACM_BEGTIM]);
;  679          1715  5                REC[SMQ$L_ACM_GETCNT] = 0;
;  680          1716  5                REC[SMQ$L_ACM_PAGECNT] = 0;
;  681          1717  5                REC[SMQ$L_ACM_QIOCNT] = 0;
;  682          1718  5                REC[SMQ$L_ACM_SYMCPUTIM] = 0;
;  683          1719  5                REC[SMQ$L_STREAM_SCT] = 0;
;  684          1720  5                REC[SMQ$W_OPEN_JOB_COUNT] = 0;
;  685          1721  5                REC[SMQ$W_PENDING_JOB_COUNT] = 0;
;  686          1722  5                REC[SMQ$W_TIMER_JOB_COUNT] = 0;
;  687          1723  5                REC[SMQ$B_CURRENT_JOB_COUNT] = 0;
;  688          1724  5                REC[SMQ$B_STREAM_INDEX] = 0;
;  689          1725  4                END;
;  690          1726  4
;  691          1727  4
;  692          1728  4            MODIFIED = TRUE;
;  693          1729  3            END;
;  694          1730  3
;  695          1731  3
;  696          1732  3        [SYM$K_SJH]:
;  697          1733  4            BEGIN
;  698          1734  4            LOCAL
;  699          1735  4                SMQ_N,                          ! Record number of job's SMQ
;  700          1736  4                SMQ:            REF BBLOCK;     ! Pointer to job's SMQ
;  701          1737  4
;  702          1738  4
;  703          1739  4            ! Validate the job's SMQ pointer and read the record.
;  704          1740  4            !
;  705          1741  4            SMQ_N = .REC[SJH$L_QUEUE_LINK];
;  706          1742  4            VALIDATE_REQUIRED_LINK(.SMQ_N, SYM$K_SMQ);
;  707          1743  4            SMQ = READ_RECORD(.SMQ_N);
;  708          1744  4
;  709          1745  4
;  710          1746  4            ! If the job was executing, write an accounting record and
;  711          1747  4            ! determine whether it is to be deleted, requeued, or retained.
;  712          1748  4            !
;  713          1749  4            IF .REC[SJH$V_EXECUTING]
;  714          1750  4            THEN
;  715          1751  5                BEGIN
;  716          1752  5                IF NOT .REC[SJH$V_RESTART]
;  717          1753  5                THEN
;  718          1754  5                    IF NOT .SMQ[SMQ$V_RETAIN_ALL_JOBS]
;  719          1755  5                    AND NOT .SMQ[SMQ$V_RETAIN_ERROR_JOBS]
;  720          1756  5                    THEN
;  721          1757  5                        DELETE = TRUE
;  722          1758  5                    ELSE
```

```
;  723          1759  5                    REC[SJH$V_RETAINED] = TRUE;
;  724          1760  5
;  725          1761  5
;  726          1762  5                REC[SJH$V_RESTARTING] = TRUE;
;  727          1763  5                REC[SJH$L_CONDITION_1] = JBC$_SYSFAIL OR STS$K_ERROR;
;  728          1764  5                REC[SJH$L_CONDITION_2] = 0;
;  729          1765  5                REC[SJH$L_CONDITION_3] = 0;
;  730          1766  5                WRITE_ACCOUNTING_RECORD(
;  731          1767  5                    .REC, .SMQ,
;  732          1768  5                    0,
;  733          1769  5                    .REC[SJH$L_CONDITION_1]);
;  734          1770  4                END;
;  735          1771  4
;  736          1772  4
;  737          1773  4            ! Reinitialize fields of the job header.
;  738          1774  4            !
;  739          1775  4            REC[SJH$L_CURRENT_FILE_LINK] = 0;
;  740          1776  4            REC[SJH$V_ABORTED] = FALSE;
;  741          1777  4            REC[SJH$V_ABORTING] = FALSE;
;  742          1778  4            REC[SJH$V_EXECUTING] = FALSE;
;  743          1779  4            REC[SJH$V_FILE_STARTING] = FALSE;
;  744          1780  4            REC[SJH$V_OPEN] = FALSE;
;  745          1781  4            REC[SJH$V_REQUEUE] = FALSE;
;  746          1782  4            REC[SJH$V_REQUEUE_HOLD] = FALSE;
;  747          1783  4            REC[SJH$V_STARTING] = FALSE;
;  748          1784  4            REC[SJH$V_SYNCHRONIZE] = FALSE;
;  749          1785  4            REC[SJH$V_SYSTEM_FAILURE] = FALSE;
;  750          1786  4            REC[SJH$L_EXECUTOR_PID] = 0;
;  751          1787  4            REC[SJH$L_REQUEUE_QUEUE_LINK] = 0;
;  752          1788  4            REC[SJH$B_REQUEUE_PRIORITY] = 0;
;  753          1789  4
;  754          1790  4
;  755          1791  4            ! Traverse substructures of the job header.
;  756          1792  4            !
;  757        P 1793  4            MARK_VARIABLE_DATA(
;  758        P 1794  4                SJH$S_CHECKPOINT, REC[SJH$T_CHECKPOINT],
;  759          1795  4                SYM$K_CHECKPOINT, .DELETE);
;  760          1796  4            TREE(.REC[SJH$L_FILE_LIST], SYM$K_SQR, .DELETE);
;  761        P 1797  4            VALIDATE_OPTIONAL_LINK(
;  762          1798  4                .REC[SJH$L_FORM_LINK], SYM$K_SFM);
;  763        P 1799  4            VALIDATE_OPTIONAL_LINK(
;  764          1800  4                .REC[SJH$L_LOG_QUEUE_LINK], SYM$K_SMQ);
;  765        P 1801  4            MARK_VARIABLE_DATA(
;  766        P 1802  4                SJH$S_LOG_SPECIFICATION, REC[SJH$T_LOG_SPECIFICATION],
;  767          1803  4                SYM$K_LOG_SPECIFICATION, .DELETE);
;  768        P 1804  4            MARK_VARIABLE_DATA(
;  769        P 1805  4                SJH$S_NOTE, REC[SJH$T_NOTE],
;  770          1806  4                SYM$K_NOTE, .DELETE);
;  771        P 1807  4            MARK_VARIABLE_DATA(
;  772        P 1808  4                SJH$S_OPERATOR_REQUEST, REC[SJH$T_OPERATOR_REQUEST],
;  773          1809  4                SYM$K_OPERATOR_REQUEST, .DELETE);
;  774        P 1810  4            MARK_VARIABLE_DATA(
;  775        P 1811  4                SJH$S_PARAMETERS, REC[SJH$T_PARAMETERS],
;  776          1812  4                SYM$K_PARAMETERS, .DELETE);
;  777        P 1813  4            MARK_VARIABLE_DATA(
;  778        P 1814  4                SJH$S_REFUSAL_REASON, REC[SJH$T_REFUSAL_REASON],
;  779          1815  4                SYM$K_REFUSAL_REASON, .DELETE);
```

```
;  780          1816  4
;  781          1817  4
;  782          1818  4              ! If the job is not to be deleted, reference the entry number and
;  783          1819  4              ! enqueue the job.
;  784          1820  4              !
;  785          1821  4              IF .DELETE
;  786          1822  4              THEN
;  787          1823  4                  RELEASE_RECORD(.SMQ_N)
;  788          1824  4              ELSE
;  789          1825  5                  BEGIN
;  790          1826  5                  REFERENCE_ENTRY_NUMBER(.REC[SYM$L_ENTRY_NUMBER]);
;  791          1827  5                  IF NOT .TYPE[V_NO_QUEUE] THEN ENQUEUE(.SMQ, .REC_N, .REC);
;  792          1828  5                  REWRITE_RECORD(.SMQ_N);
;  793          1829  4                  END;
;  794          1830  4
;  795          1831  4
;  796          1832  4              MODIFIED = TRUE;
;  797          1833  3              END;
;  798          1834  3
;  799          1835  3
;  800          1836  3          [SYM$K_SQR]:
;  801          1837  4              BEGIN
;  802          1838  4
;  803          1839  4              ! Traverse substructures of the file record.
;  804          1840  4              !
;  805        P 1841  4              MARK_VARIABLE_DATA(
;  806        P 1842  4                  SQR$S_FILE_SETUP_MODULES, REC[SQR$T_FILE_SETUP_MODULES],
;  807          1843  4                  SYM$K_FILE_SETUP_MODULES, .DELETE);
;  808          1844  3              END;
;  809          1845  3
;  810          1846  3
;  811          1847  3          [SYM$K_ENTRY_BITMAP]:
;  812          1848  4              BEGIN
;  813          1849  4
;  814          1850  4              ! Reinitialize the bitmap.
;  815          1851  4              !
;  816          1852  4              CH$FILL(0, SYM$S_DATA, REC[SYM$T_DATA]);
;  817          1853  4              MODIFIED = TRUE;
;  818          1854  3              END;
;  819          1855  3
;  820          1856  3
;  821          1857  3          [SYM$K_GENERIC_TARGET]:
;  822          1858  4              BEGIN
;  823          1859  4
;  824          1860  4              ! Validate count.
;  825          1861  4              !
;  826          1862  4              IF .VECTOR[REC[SYM$T_DATA], 0] GTRU 124
;  827          1863  4              THEN
;  828          1864  4                  VECTOR[REC[SYM$T_DATA], 0] = 124;
;  829          1865  4
;  830          1866  4
;  831          1867  4              ! Validate queue pointers.
;  832          1868  4              !
;  833          1869  4              INCR N FROM 1 TO .VECTOR[REC[SYM$T_DATA], 0] DO
;  834          1870  5                  BEGIN
;  835        P 1871  5                  VALIDATE_REQUIRED_LINK(
;  836          1872  5                      .VECTOR[REC[SYM$T_DATA], .N], SYM$K_SMQ);
```

```
;  837          1873  4                    END;
;  838          1874  3                END;
;  839          1875  3
;  840          1876  3
;  841          1877  3            TES;
;  842          1878  3
;  843          1879  3
;  844          1880  3        ! If the record is to be deleted, link it to the free list.
;  845          1881  3        !
;  846          1882  3        IF .DELETE
;  847          1883  3        THEN
;  848          1884  4            BEGIN
;  849          1885  4            RTM[.REC_N-1] = 0;
;  850          1886  4            CH$COPY(4, SQH[SQH$L_FREE_LIST], 0, SYM$S_SYM, .REC);
;  851          1887  4            SQH[SQH$L_FREE_LIST] = .REC_N;
;  852          1888  4            MODIFIED = TRUE;
;  853          1889  3            END;
;  854          1890  3
;  855          1891  3
;  856          1892  3        ! If the record was modified, write it back; otherwise release it.
;  857          1893  3        !
;  858          1894  3        IF .MODIFIED
;  859          1895  3            THEN REWRITE_RECORD(.REC_N)
;  860          1896  3            ELSE RELEASE_RECORD(.REC_N);
;  861          1897  3
;  862          1898  3
;  863          1899  3        ! Advance to next record.
;  864          1900  3        !
;  865          1901  3        IF .TYPE[V_NO_TRAVERSE] THEN EXITLOOP;
;  866          1902  3        REC_N = .REC_NS;
;  867          1903  2        END;
;  868          1904  1 END;


                                                03FC 00000 TREE:    .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9                 ; 1474
                                              5E        04 C2 00002         SUBL2   #4, SP                                  ;
                                              57    04  AC D0 00005         MOVL    RECNO, REC_N                            ; 1562
                                    7E    09  AC        01 00 EF 00009     EXTZV   #0, #1, TYPE+1, -(SP)                   ; 1599
                                              6E        6E D2 0000F         MCOML   (SP), (SP)                              ;
                                                        57 D5 00012 1$:     TSTL    REC_N                                   ; 1563
                                                        01 12 00014         BNEQ    2$                                      ;
                                                           04 00016         RET                                             ;
                                    00000000'             57 D0 00017 2$:   MOVL    REC_N, DIAG_TRACE+36                    ; 1572
                                              02          57 D1 0001E         CMPL    REC_N, #2                               ; 1575
                                                        06 1F 00021         BLSSU   3$                                      ;
                                          40  AB        57 D1 00023         CMPL    REC_N, 64(SQH)                          ; 1576
                                                        0D 1B 00027         BLEQU   4$                                      ;
                                                00048440 8F DD 00029 3$:    PUSHL   #296000                                 ; 1578
                                    00000000G 00        01 FB 0002F         CALLS   #1, LIB$SIGNAL                          ;
                                                        57 DD 00036 4$:     PUSHL   REC_N                                   ; 1583
                                    00000000G EF        01 FB 00038         CALLS   #1, READ_RECORD                         ;
                                              56        50 D0 0003F         MOVL    R0, REC                                 ;
                                                        59 D4 00042         CLRL    MODIFIED                                ; 1584
                                                  58    0C AC D0 00044      MOVL    DEL, DELETE                             ; 1585
```

```
                04   AE               66  D0 00048        MOVL    (REC), REC_NS                    ; 1586
                08   AC          04   A6  91 0004C        CMPB    4(REC), TYPE                     ; 1591
                                      0D  13 00051        BEQL    5$
                           00048440   8F  DD 00053        PUSHL   #296000                          ; 1593
          00000000G  00               01  FB 00059        CALLS   #1, LIB$SIGNAL
                     19               6E  E9 00060 5$:    BLBC    (SP), 7$                         ; 1599
                            FF A74A   95 00063            TSTB    -1(REC_N)[RTM]                   ; 1602
                                      0D  19 00067        BLSS    6$
                           00048440   8F  DD 00069        PUSHL   #296000
          00000000G  00               01  FB 0006F        CALLS   #1, LIB$SIGNAL
             FF A74A             04   A6  90 00076 6$:    MOVB    4(REC), -1(REC_N)[RTM]           ; 1603
                  14 01          04   A6  8F 0007C 7$:    CASEB   4(REC), #1, #20                  ; 1609
0344      0051      002C        0344         00081 8$:    .WORD   52$-8$,-
02E3      014E      0096        006F         00089                9$-8$,-
0344      0344      02FA        0344         00091                11$-8$,-
0344      0344      0308        0344         00099                52$-8$,-
0344      0344      0344        0344         000A1                13$-8$,-
                                0344         000A9                16$-8$,-
                                                                  25$-8$,-
                                                                  43$-8$,-
                                                                  52$-8$,-
                                                                  45$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$,-
                                                                  47$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$,-
                                                                  52$-8$
                                      67  11 000AB        BRB     15$
                     50        015D   C6  9E 000AD 9$:    MOVAB   349(REC), R0                     ; 1624
                     04               60  B1 000B2        CMPW    (R0), #4
                                      0B  1B 000B5        BLEQU   10$
                     7E               0D  7D 000B7        MOVQ    #13, -(SP)
                                 02   A0  DD 000BA        PUSHL   2(R0)
                FF3E CF               03  FB 000BD        CALLS   #3, TREE
                     50        0163   C6  9E 000C2 10$:   MOVAB   355(REC), R0                     ; 1627
                     04               60  B1 000C7        CMPW    (R0), #4
                                      48  1B 000CA        BLEQU   15$
                     7E               13  7D 000CC        MOVQ    #19, -(SP)
                                    029F  31 000CF        BRW     44$
                     52          0C   A6  9E 000D2 11$:   MOVAB   12(R6), SFE                      ; 1636
                                      53  D4 000D6        CLRL    SFE_N                            ; 1637
                                      62  95 000D8 12$:   TSTB    (SFE)                            ; 1639
                                      38  13 000DA        BEQL    15$
                     7E               02  7D 000DC        MOVQ    #2, -(SP)                        ; 1644
                                 24   A2  DD 000DF        PUSHL   36(SFE)
                FF19 CF               03  FB 000E2        CALLS   #3, TREE
                     52               28  C0 000E7        ADDL2   #40, SFE                         ; 1645
          EA         53               0B  F3 000EA        AOBLEQ  #11, SFE_N, 12$                  ; 1637
                                      24  11 000EE        BRB     15$                              ; 1609
                     54          08   AC  D0 000F0 13$:   MOVL    TYPE, T                          ; 1657
                     54               06  90 000F4        MOVB    #6, T
```

```
                        52        0C    A6  9E 000F7          MOVAB   12(R6), SQE                     ; 1658
                                        53  D4 000FB          CLRL    SQE_N                           ; 1659
                                        62  95 000FD 14$:      TSTB    (SQE)                           ; 1661
                                        13  13 000FF          BEQL    15$                             ;
                                        7E  D4 00101          CLRL    -(SP)                           ; 1666
                                        54  DD 00103          PUSHL   T                               ;
                                    24  A2  DD 00105          PUSHL   36(SQE)                         ;
                              FEF3  CF      03  FB 00108      CALLS   #3, TREE                        ;
                                    52      28  C0 0010D      ADDL2   #40, SQE                        ; 1667
                          E9        53      0B  F3 00110      AOBLEQ  #11, SQE_N, 14$                 ; 1659
                                        02AE  31  00114 15$:  BRW     52$                             ; 1609
                          10    09  AC      04  E1 00117 16$: BBC     #4, TYPE+1, 17$                 ; 1680
                                    7E      07  7D 0011C      MOVQ    #7, -(SP)                       ; 1683
                                    48  A6  DD 0011F          PUSHL   72(REC)                         ;
                              FED9  CF      03  FB 00122      CALLS   #3, TREE                        ;
                                    48  A6  7C 00127          CLRQ    72(REC)                         ; 1684
                                        7E  11 0012A          BRB     22$                             ; 1680
                          7B    09  AC      03  E1 0012C 17$: BBC     #3, TYPE+1, 23$                 ; 1692
                                    50  2C  A6  D0 00131      MOVL    44(REC), R0                     ; 1696
                                        1F  13 00135          BEQL    19$                             ;
                                    02      50  D1 00137      CMPL    R0, #2                          ;
                                        0D  1F 0013A          BLSSU   18$                             ;
                                40  AB      50  D1 0013C      CMPL    R0, 64(SQH)                     ;
                                        07  1A 00140          BGTRU   18$                             ;
                                    06  FF  A04A  91 00142    CMPB    -1(R0)[RTM], #6                 ;
                                        0D  13 00147          BEQL    19$                             ;
                                00048440    8F  DD 00149 18$: PUSHL   #296000                         ;
                      00000000G 00          01  FB 0014F      CALLS   #1, LIB$SIGNAL                  ;
                                    50  70  A6  D0 00156 19$: MOVL    112(REC), R0                    ; 1698
                                        1F  13 0015A          BEQL    21$                             ;
                                    02      50  D1 0015C      CMPL    R0, #2                          ;
                                        0D  1F 0015F          BLSSU   20$                             ;
                                40  AB      50  D1 00161      CMPL    R0, 64(SQH)                     ;
                                        07  1A 00165          BGTRU   20$                             ;
                                    02  FF  A04A  91 00167    CMPB    -1(R0)[RTM], #2                 ;
                                        0D  13 0016C          BEQL    21$                             ;
                                00048440    8F  DD 0016E 20$: PUSHL   #296000                         ;
                      00000000G 00          01  FB 00174      CALLS   #1, LIB$SIGNAL                  ;
                                    7E      0E  7D 0017B 21$: MOVQ    #14, -(SP)                      ; 1699
                                        74  A6  DD 0017E      PUSHL   116(REC)                        ;
                              FE7A  CF      03  FB 00181      CALLS   #3, TREE                        ;
                                    7E      7E  D4 00186      CLRL    -(SP)                           ; 1700
                                    7E  0207    8F  3C 00188  MOVZWL  #519, -(SP)                     ;
                                        78  A6  DD 0018D      PUSHL   120(REC)                        ;
                              FE6B  CF      03  FB 00190      CALLS   #3, TREE                        ;
                                    50  0118  C6  9E 00195    MOVAB   280(REC), R0                    ; 1703
                                    04      60  B1 0019A      CMPW    (R0), #4                        ;
                                        2D  1B 0019D          BLEQU   24$                             ;
                                    7E      0F  7D 0019F      MOVQ    #15, -(SP)                      ;
                                        02  A0  DD 001A2      PUSHL   2(R0)                           ;
                              FE56  CF      03  FB 001A5      CALLS   #3, TREE                        ;
                                        20  11 001AA 22$:     BRB     24$                             ; 1692
                                        10  A6  D4 001AC 23$: CLRL    16(REC)                         ; 1712
                                11  A6      02  88 001AF      BISB2   #2, 17(REC)                     ; 1713
                                        14  A6  7C 001B3      CLRQ    20(REC)                         ; 1714
                                        1C  A6  7C 001B6      CLRQ    28(REC)                         ; 1715
                                        24  A6  7C 001B9      CLRQ    36(REC)                         ; 1717
```

```
                        00FC    C6  7C 001BC          CLRQ    252(REC)                                    ; 1719
                        010C    C6  B4 001C0          CLRW    268(REC)                                    ; 1722
                        0115    C6  94 001C4          CLRB    277(REC)                                    ; 1723
                        0117    C6  94 001C8          CLRB    279(REC)                                    ; 1724
                             01B5   31 001CC 24$:      BRW     46$                                         ; 1728
                    54      0134    C6  D0 001CF 25$:  MOVL    308(REC), SMQ_N                             ; 1741
                    02              54  D1 001D4       CMPL    SMQ_N, #2                                   ; 1742
                                    0D  1F 001D7       BLSSU   26$                                         ;
                40  AB              54  D1 001D9       CMPL    SMQ_N, 64(SQH)                              ;
                                    07  1A 001DD       BGTRU   26$                                         ;
                    06          FF A44A 91 001DF       CMPB    -1(SMQ_N)[RTM], #6                          ;
                                    0D  13 001E4       BEQL    27$                                         ;
                        00048440    8F  DD 001E6 26$:  PUSHL   #296000                                     ;
          00000000G 00              01  FB 001EC       CALLS   #1, LIB$SIGNAL                              ;
                                    54  DD 001F3 27$:  PUSHL   SMQ_N                                      ; 1743
          00000000G EF              01  FB 001F5       CALLS   #1, READ_RECORD                             ;
                    53              50  D0 001FC       MOVL    R0, SMQ                                     ;
                    52          10  A6  9E 001FF       MOVAB   16(REC), R2                                 ; 1749
          3A        62              03  E1 00203       BBC     #3, (R2), 30$                               ;
          13    0E  A6              01  E0 00207       BBS     #1, 14(REC), 29$                            ; 1752
          04    0E  A3              02  E0 0020C       BBS     #2, 14(SMQ), 28$                            ; 1754
          05    0E  A3              03  E0 00211       BBS     #3, 14(SMQ), 28$                            ; 1755
                    58              01  D0 00216       MOVL    #1, DELETE                                  ; 1757
                                    04  11 00219       BRB     29$                                         ;
                01  A2              08  88 0021B 28$:  BISB2   #8, 1(R2)                                  ; 1759
                01  A2              04  88 0021F 29$:  BISB2   #4, 1(R2)                                  ; 1762
              00DC  C6  000480F2    8F  D0 00223       MOVL    #295154, 220(REC)                           ; 1763
                        00E0    C6  7C 0022C       CLRQ    224(REC)                                    ; 1764
                        00DC    C6  DD 00230       PUSHL   220(REC)                                    ; 1769
                                7E  D4 00234       CLRL    -(SP)                                       ; 1766
                                53  DD 00236       PUSHL   SMQ                                         ; 1767
                                56  DD 00238       PUSHL   REC                                         ;
00000000G       EF              04  FB 0023A       CALLS   #4, WRITE_ACCOUNTING_RECORD                 ;
                        00F0    C6  D4 00241 30$:  CLRL    240(REC)                                    ; 1775
                    62  735B    8F  AA 00245       BICW2   #29531, (R2)                                ; 1785
                        0168    C6  D4 0024A       CLRL    360(REC)                                    ; 1786
                        0138    C6  D4 0024E       CLRL    312(REC)                                    ; 1787
                        017E    C6  94 00252       CLRB    382(REC)                                    ; 1788
                    50  0180    C6  9E 00256       MOVAB   384(REC), R0                                ; 1795
                    1E              60  B1 0025B       CMPW    (R0), #30                                   ;
                                    0C  1B 0025E       BLEQU   31$                                         ;
                                    58  DD 00260       PUSHL   DELETE                                      ;
                                    0B  DD 00262       PUSHL   #11                                         ;
                            02      A0  DD 00264       PUSHL   2(R0)                                       ;
                FD94        CF          03  FB 00267       CALLS   #3, TREE                                    ;
                                    58  DD 0026C 31$:  PUSHL   DELETE                                      ; 1796
                                    08  DD 0026E       PUSHL   #8                                          ;
                        00F4    C6  DD 00270       PUSHL   244(REC)                                    ;
                FD87    CF          03  FB 00274       CALLS   #3, TREE                                    ;
                    50  00FC    C6  D0 00279       MOVL    252(REC), R0                                ; 1798
                                1F  13 0027E       BEQL    33$                                         ;
                    02              50  D1 00280       CMPL    R0, #2                                      ;
                                    0D  1F 00283       BLSSU   32$                                         ;
                40  AB              50  D1 00285       CMPL    R0, 64(SQH)                                 ;
                                    07  1A 00289       BGTRU   32$                                         ;
                    02          FF A04A 91 0028B       CMPB    -1(R0)[RTM], #2                             ;
                                    0D  13 00290       BEQL    33$                                         ;
```

```
                        00048440   8F   DD 00292 32$:    PUSHL   #296000                                        ;
                00000000G  00      01   FB 00298         CALLS   #1, LIB$SIGNAL                                 ;
                           50 0104 C6   D0 0029F 33$:    MOVL    260(REC), R0                                   ; 1800
                                   1F   13 002A4         BEQL    35$                                            ;
                           02           50   D1 002A6    CMPL    R0, #2                                         ;
                                        0D   1F 002A9    BLSSU   34$                                            ;
                        40 AB           50   D1 002AB    CMPL    R0, 64(SQH)                                    ;
                                        07   1A 002AF    BGTRU   34$                                            ;
                           06       FF A04A 91   002B1    CMPB    -1(R0)[RTM], #6                                ;
                                        0D   13 002B6    BEQL    35$                                            ;
                        00048440   8F   DD 002B8 34$:    PUSHL   #296000                                        ;
                00000000G  00      01   FB 002BE         CALLS   #1, LIB$SIGNAL                                 ;
                           50 01A0 C6   9E 002C5 35$:    MOVAB   416(REC), R0                                   ; 1803
                           04           60   B1 002CA    CMPW    (R0), #4                                       ;
                                        0C   1B 002CD    BLEQU   36$                                            ;
                                        58   DD 002CF    PUSHL   DELETE                                         ;
                                        10   DD 002D1    PUSHL   #16                                            ;
                              02        A0   DD 002D3    PUSHL   2(R0)                                          ;
                        FD25  CF        03   FB 002D6    CALLS   #3, TREE                                       ;
                           50 01A6 C6   9E 002DB 36$:    MOVAB   422(REC), R0                                   ; 1806
                           04           60   B1 002E0    CMPW    (R0), #4                                       ;
                                        0C   1B 002E3    BLEQU   37$                                            ;
                                        58   DD 002E5    PUSHL   DELETE                                         ;
                                        11   DD 002E7    PUSHL   #17                                            ;
                              02        A0   DD 002E9    PUSHL   2(R0)                                          ;
                        FD0F  CF        03   FB 002EC    CALLS   #3, TREE                                       ;
                           50 01AC C6   9E 002F1 37$:    MOVAB   428(REC), R0                                   ; 1809
                           04           60   B1 002F6    CMPW    (R0), #4                                       ;
                                        0C   1B 002F9    BLEQU   38$                                            ;
                                        58   DD 002FB    PUSHL   DELETE                                         ;
                                        12   DD 002FD    PUSHL   #18                                            ;
                              02        A0   DD 002FF    PUSHL   2(R0)                                          ;
                        FCF9  CF        03   FB 00302    CALLS   #3, TREE                                       ;
                           50 01B2 C6   9E 00307 38$:    MOVAB   434(REC), R0                                   ; 1812
                           1E           60   B1 0030C    CMPW    (R0), #30                                      ;
                                        0C   1B 0030F    BLEQU   39$                                            ;
                                        58   DD 00311    PUSHL   DELETE                                         ;
                                        14   DD 00313    PUSHL   #20                                            ;
                              02        A0   DD 00315    PUSHL   2(R0)                                          ;
                        FCE3  CF        03   FB 00318    CALLS   #3, TREE                                       ;
                           50 01D2 C6   9E 0031D 39$:    MOVAB   466(REC), R0                                   ; 1815
                           04           60   B1 00322    CMPW    (R0), #4                                       ;
                                        0C   1B 00325    BLEQU   40$                                            ;
                                        58   DD 00327    PUSHL   DELETE                                         ;
                                        15   DD 00329    PUSHL   #21                                            ;
                              02        A0   DD 0032B    PUSHL   2(R0)                                          ;
                        FCCD  CF        03   FB 0032E    CALLS   #3, TREE                                       ;
                           0B           58   E9 00333 40$:    BLBC    DELETE, 41$                                ; 1821
                                        54   DD 00336    PUSHL   SMQ_N                                          ; 1823
                     00000000G  EF      01   FB 00338    CALLS   #1, RELEASE_RECORD                             ;
                                        43   11 0033F    BRB     46$                                            ;
                              08        A6   DD 00341 41$:    PUSHL   8(REC)                                     ; 1826
                        FB0E  CF        01   FB 00344    CALLS   #1, REFERENCE_ENTRY_NUMBER                     ;
                     0B    09 AC        01   E0 00349    BBS     #1, TYPE+1, 42$                                ; 1827
                                        56   DD 0034E    PUSHL   REC                                            ;
                                 0088   8F   BB 00350    PUSHR   #^M<R3,R7>                                     ;
                        FB56  CF        03   FB 00354    CALLS   #3, ENQUEUE                                    ;
```

```
                                                    54  DD 00359 42$:    PUSHL   SMQ_N                                   ; 1828
                                00000000G   EF          01  FB 0035B         CALLS   #1, REWRITE_RECORD                      ;
                                            20          11  00362         BRB     46$                                     ; 1832
                                        50      45      A6  9E 00364 43$:  MOVAB   69(REC), R0                             ; 1843
                                        04              60  B1 00368         CMPW    (R0), #4                                ;
                                                        58  1B 0036B         BLEQU   52$                                     ;
                                                        58  DD 0036D         PUSHL   DELETE                                  ;
                                                        0C  DD 0036F         PUSHL   #12                                     ;
                                            02          A0  DD 00371 44$:     PUSHL   2(R0)                                   ;
                                    FC87    CF          03  FB 00374         CALLS   #3, TREE                                ;
                                                        4A  11 00379         BRB     52$                                     ; 1609
    01F4    8F          00              6E              00  2C 0037B 45$:   MOVC5   #0, (SP), #0, #500, 12(REC)            ; 1852
                                            0C          A6      00382                                                         ;
                                        59              01  D0 00384 46$:    MOVL    #1, MODIFIED                            ; 1853
                                                        3C  11 00387         BRB     52$                                     ; 1609
                                0000007C    8F      0C  A6  D1 00389 47$:    CMPL    12(REC), #124                           ; 1862
                                                        05  1B 00391         BLEQU   48$                                     ;
                                    0C      A6      7C  8F  9A 00393         MOVZBL  #124, 12(REC)                           ; 1864
                                                        52  D4 00398 48$:    CLRL    N                                       ; 1869
                                                        24  11 0039A         BRB     51$                                     ;
                                        50      0C A642  D0 0039C 49$:       MOVL    12(REC)[N], R0                          ; 1872
                                        02              50  D1 003A1         CMPL    R0, #2                                  ;
                                                        0D  1F 003A4         BLSSU   50$                                     ;
                                    40  AB              50  D1 003A6         CMPL    R0, 64(SQH)                             ;
                                                        07  1A 003AA         BGTRU   50$                                     ;
                                        06      FF  A04A  91 003AC         CMPB    -1(R0)[RTM], #6                         ;
                                                        0D  13 003B1         BEQL    51$                                     ;
                                    00048440        8F  DD 003B3 50$:        PUSHL   #296000                                 ;
                                00000000G   00          01  FB 003B9         CALLS   #1, LIB$SIGNAL                          ;
                                D7              52      0C  A6  F3 003C0 51$:    AOBLEQ  12(REC), N, 49$                     ; 1869
                                                14      58  E9 003C5 52$:    BLBC    DELETE, 53$                             ; 1882
                                            FF  A74A    94  003C8            CLRB    -1(REC_N)[RTM]                          ; 1885
    0200    8F          00          38      04  2C 003CC         MOVC5   #4, 56(SQH), #0, #512, (REC)                    ; 1886
                                                66      003D4                                                               ;
                                        38      57  D0 003D5         MOVL    REC_N, 56(SQH)                                  ; 1887
                                        59      01  D0 003D9         MOVL    #1, MODIFIED                                    ; 1888
                                                0B  59  E9 003DC 53$:    BLBC    MODIFIED, 54$                               ; 1894
                                                    57  DD 003DF         PUSHL   REC_N                                       ; 1895
                                00000000G   EF          01  FB 003E1         CALLS   #1, REWRITE_RECORD                      ;
                                            09          11  003E8         BRB     55$                                     ;
                                                        57  DD 003EA 54$:    PUSHL   REC_N                                   ; 1896
                                00000000G   EF          01  FB 003EC         CALLS   #1, RELEASE_RECORD                      ;
                                    07      09      AC  02  E0 003F3 55$:    BBS     #2, TYPE+1, 56$                         ; 1901
                                            57      04  AE  D0 003F8         MOVL    REC_NS, REC_N                           ; 1902
                                                    FC13    31  003FC         BRW     1$                                      ; 1563
                                                            04  003FF 56$:    RET                                             ; 1904

; Routine Size:  1024 bytes,    Routine Base:  CODE + 01B5
```

```
;  870          1905  1 ROUTINE COLD_START_CURRENT_LIST: NOVALUE=
;  871          1906  1
;  872          1907  1 !++
;  873          1908  1 !
;  874          1909  1 ! FUNCTIONAL DESCRIPTION:
;  875          1910  1 !       This routine scans each executor queue's CURRENT_LIST looking
;  876          1911  1 !       for deallocated SJH records.  If any are found they are unlinked
;  877          1912  1 !       from the CURRENT_LIST.
;  878          1913  1 !
;  879          1914  1 !       This routine is called just before the third pass of the queue
;  880          1915  1 !       index list.
;  881          1916  1 !
;  882          1917  1 ! INPUT PARAMETERS:
;  883          1918  1 !       None.
;  884          1919  1 !
;  885          1920  1 ! IMPLICIT INPUTS:
;  886          1921  1 !       None.
;  887          1922  1 !
;  888          1923  1 ! OUTPUT PARAMETERS:
;  889          1924  1 !       None.
;  890          1925  1 !
;  891          1926  1 ! IMPLICIT OUTPUTS:
;  892          1927  1 !       None.
;  893          1928  1 !
;  894          1929  1 ! ROUTINE VALUE:
;  895          1930  1 !       None.
;  896          1931  1 !
;  897          1932  1 ! SIDE EFFECTS:
;  898          1933  1 !       None.
;  899          1934  1 !
;  900          1935  1 !--
;  901          1936  1
;  902          1937  2 BEGIN
;  903          1938  2
;  904          1939  2 LOCAL
;  905          1940  2         SQX_N,                          ! Record Number of Queue Index
;  906          1941  2         SQH:    REF BBLOCK;             ! Queue Header Record
;  907          1942  2
;  908          1943  2 ! Read queue header record.
;  909          1944  2 !
;  910          1945  2 SQH = READ_RECORD(SQH$K_RECNO);
;  911          1946  2
;  912          1947  2 ! Validate each Queue's CURRENT_LIST.
;  913          1948  2 !
;  914          1949  2 SQX_N = .SQH[SQH$L_QUEUE_INDEX_LIST];
;  915          1950  2 WHILE .SQX_N NEQ 0 DO
;  916          1951  3     BEGIN
;  917          1952  3
;  918          1953  3     LOCAL
;  919          1954  3             SQX_NS,                     ! Successor of Queue Index
;  920          1955  3             SQX:        REF BBLOCK,     ! Queue Index Record
;  921          1956  3             SQE_N,                      ! Queue Entry Number
;  922          1957  3             SQE:        REF BBLOCK;     ! Queue Entry
;  923          1958  3
;  924          1959  3     ! Read the queue index record.
;  925          1960  3     !
;  926          1961  3     SQX = READ_RECORD(.SQX_N);
```

```
;  927          1962  3
;  928          1963  3
;  929          1964  3      ! Scan the queue index record.
;  930          1965  3      !
;  931          1966  3      SQE = SQX[SYM$T_DATA];
;  932          1967  3      INCR SQE_N FROM 0 TO SQX$K_ENTRIES-1 DO
;  933          1968  4          BEGIN
;  934          1969  4          IF CH$RCHAR(SQE[SQX$T_NAME]) EQL 0
;  935          1970  4          THEN
;  936          1971  4              EXITLOOP
;  937          1972  4          ELSE
;  938          1973  5              BEGIN
;  939          1974  5              IF .SQE[SQX$V_EXECUTOR]
;  940          1975  5              THEN
;  941          1976  6                  BEGIN
;  942          1977  6
;  943          1978  6                  LOCAL
;  944          1979  6                          SMQ:    REF BBLOCK,     ! Queue Record
;  945          1980  6                          SMQ_N;                  ! Queue Record Number
;  946          1981  6
;  947          1982  6                  SMQ = READ_RECORD(SMQ_N = .SQE[SQX$L_QUEUE_LINK]);
;  948          1983  6
;  949          1984  6                  ! Scan the current list.  Verify that any jobs on the list
;  950          1985  6                  ! have valid back pointers to the SMQ record.  Unlink any
;  951          1986  6                  ! that does not.
;  952          1987  6                  !
;  953          1988  6                  IF .SMQ[SMQ$L_CURRENT_LIST] NEQ 0
;  954          1989  6                  THEN
;  955          1990  7                      BEGIN
;  956          1991  7                      LOCAL
;  957          1992  7                              SJH:        REF BBLOCK,     ! Job Record
;  958          1993  7                              SJH_N,                      ! " Number
;  959          1994  7                              SJH_P:      REF BBLOCK,     ! Predecessor
;  960          1995  7                              SJH_NP;                     ! "" Record Number
;  961          1996  7
;  962          1997  7                      ! Initialize predecessor to SMQ.
;  963          1998  7                      !
;  964          1999  7                      SJH_NP = .SMQ_N;
;  965          2000  7                      SJH_P = .SMQ;
;  966          2001  7
;  967          2002  7                      SJH_N = .SMQ[SMQ$L_CURRENT_LIST];
;  968          2003  7                      WHILE .SJH_N NEQ 0 DO
;  969          2004  8                          BEGIN
;  970          2005  8
;  971          2006  8                          ! Verify each SJH.
;  972          2007  8                          !
;  973          2008  8                          SJH = READ_RECORD(.SJH_N);
;  974          2009  8                          IF .SJH[SJH$L_QUEUE_LINK] EQL 0
;  975          2010  8                          THEN
;  976          2011  9                              BEGIN
;  977          2012  9
;  978          2013  9                              ! Unlink deallocated SJH from CURRENT_LIST.
;  979          2014  9                              !
;  980          2015  9                              IF .SJH_NP EQL .SMQ
;  981          2016  9                              THEN
;  982          2017 10                                  BEGIN
;  983          2018 10
```

```
;  984          2019 10                         ! Predecessor is SMQ.
;  985          2020 10                         !
;  986          2021 10                         SMQ[SMQ$L_CURRENT_LIST] = 0;
;  987          2022 10                         SMQ[SMQ$L_CURRENT_LIST_END] = 0;
;  988          2023 10                         FLUSH_RECORD(.SMQ_N);
;  989          2024 10                         END
;  990          2025  9                     ELSE
;  991          2026 10                         BEGIN
;  992          2027 10
;  993          2028 10                         ! Predecessor is previous SJH.  First update
;  994          2029 10                         ! SMQ's current list end, then unlink bad SJH.
;  995          2030 10                         !
;  996          2031 10                         SMQ[SMQ$L_CURRENT_LIST_END] = .SJH_NP;
;  997          2032 10                         FLUSH_RECORD(.SMQ_N);
;  998          2033 10                         SJH_P[SYM$L_LINK] = 0;
;  999          2034 10                         FLUSH_RECORD(.SJH_NP);
; 1000          2035  9                         END;
; 1001          2036  9
; 1002          2037  9                     EXITLOOP;
; 1003          2038  8                     END;
; 1004          2039  8
; 1005          2040  8                 ! Set up for next iteration.  Current SJH becomes
; 1006          2041  8                 ! predecessor, next SJH read will be successor of
; 1007          2042  8                 ! current SJH, if one exists.
; 1008          2043  8                 !
; 1009          2044  8                 SJH_P = .SJH;
; 1010          2045  8                 SJH_NP = .SJH_N;
; 1011          2046  8                 SJH_N = .SJH_P[SYM$L_LINK];
; 1012          2047  8                 END
; 1013          2048  7             END
; 1014          2049  6         ELSE
; 1015          2050  6             RELEASE_RECORD(.SMQ_N);
; 1016          2051  5             END;
; 1017          2052  4         END;
; 1018          2053  4
; 1019          2054  4
; 1020          2055  4         SQE = .SQE + SQX$S_SQX;
; 1021          2056  3         END;
; 1022          2057  3
; 1023          2058  3
; 1024          2059  3     ! Advance to the next index block.
; 1025          2060  3     !
; 1026          2061  3     SQX_NS = .SQX[SYM$L_LINK];
; 1027          2062  3     RELEASE_RECORD(.SQX_N);
; 1028          2063  3     SQX_N = .SQX_NS;
; 1029          2064  2     END;
; 1030          2065  1 END;


                                    0FFC 00000 COLD_START_CURRENT_LIST:
                                                       .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11          ; 1905
                                               01  DD 00002 PUSHL   #1                                       ; 1945
                        00000000G   EF                 01  FB 00004 CALLS   #1, READ_RECORD                  ;
                                    59          64     A0  D0 0000B MOVL    100(SQH), SQX_N                  ; 1949
```

```
                                        01  12 0000F 1$:    BNEQ    2$                              ; 1950
                                            04 00011        RET                                     ;
                                        59  DD 00012 2$:    PUSHL   SQX_N                           ; 1961
                00000000G               EF  01  FB 00014    CALLS   #1, READ_RECORD                 ;
                                            50  D0 0001B    MOVL    R0, SQX                         ;
                                        54      0C  AB  9E 0001E    MOVAB   12(R11), SQE            ; 1966
                                            5A  D4 00022    CLRL    SQE_N                           ; 1967
                                            64  95 00024 3$:    TSTB    (SQE)                       ; 1969
                                            79  13 00026    BEQL    10$                             ;
                    6D      20          A4  01  E1 00028    BBC     #1, 32(SQE), 9$                 ; 1974
                                        53      24  A4  D0 0002D    MOVL    36(SQE), SMQ_N          ; 1982
                                            53  DD 00031    PUSHL   SMQ_N                           ;
                00000000G               EF  01  FB 00033    CALLS   #1, READ_RECORD                 ;
                                        52      50  D0 0003A    MOVL    R0, SMQ                     ;
                                            48  A2  D5 0003D    TSTL    72(SMQ)                     ; 1988
                                            4F  13 00040    BEQL    8$                              ;
                                        55      53  D0 00042    MOVL    SMQ_N, SJH_NP               ; 1999
                                        58      52  D0 00045    MOVL    SMQ, SJH_P                  ; 2000
                                        56      48  A2  D0 00048    MOVL    72(SMQ), SJH_N          ; 2002
                                            4C  13 0004C 4$:    BEQL    9$                          ; 2003
                                            56  DD 0004E    PUSHL   SJH_N                           ; 2008
                00000000G               EF  01  FB 00050    CALLS   #1, READ_RECORD                 ;
                                        57      50  D0 00057    MOVL    R0, SJH                     ;
                                              0134  C7  D5 0005A    TSTL    308(SJH)                ; 2009
                                            26  12 0005E    BNEQ    7$                              ;
                                        52      55  D1 00060    CMPL    SJH_NP, SMQ                 ; 2015
                                            07  12 00063    BNEQ    5$                              ;
                                            48  A2  7C 00065    CLRQ    72(SMQ)                     ; 2021
                                            53  DD 00068    PUSHL   SMQ_N                           ; 2023
                                            11  11 0006A    BRB     6$                              ;
                                      4C    A2  55  D0 0006C 5$:    MOVL    SJH_NP, 76(SMQ)         ; 2031
                                            53  DD 00070    PUSHL   SMQ_N                           ; 2032
                00000000G               EF  01  FB 00072    CALLS   #1, FLUSH_RECORD                ;
                                            68  D4 00079    CLRL    (SJH_P)                         ; 2033
                                            55  DD 0007B    PUSHL   SJH_NP                          ; 2034
                00000000G               EF  01  FB 0007D 6$:    CALLS   #1, FLUSH_RECORD            ;
                                            14  11 00084    BRB     9$                              ; 2011
                                        58      57  D0 00086 7$:    MOVL    SJH, SJH_P              ; 2044
                                        55      56  D0 00089    MOVL    SJH_N, SJH_NP               ; 2045
                                        56      68  D0 0008C    MOVL    (SJH_P), SJH_N              ; 2046
                                            BB  11 0008F    BRB     4$                              ; 2003
                                            53  DD 00091 8$:    PUSHL   SMQ_N                       ; 2050
                00000000G               EF  01  FB 00093    CALLS   #1, RELEASE_RECORD              ;
                                        54      28  C0 0009A 9$:    ADDL2   #40, SQE                ; 2055
                    83                  5A  0B  F3 0009D    AOBLEQ  #11, SQE_N, 3$                  ; 1967
                                        53      6B  D0 000A1 10$:   MOVL    (SQX), SQX_NS           ; 2061
                                            59  DD 000A4    PUSHL   SQX_N                           ; 2062
                00000000G               EF  01  FB 000A6    CALLS   #1, RELEASE_RECORD              ;
                                        59      53  D0 000AD    MOVL    SQX_NS, SQX_N               ; 2063
                                              FF5C  31 000B0    BRW     1$                          ; 1950
                                            04 000B3        RET                                     ; 2065

; Routine Size:  180 bytes,    Routine Base:  CODE + 05B5
```

```
;  1032          2066  1 ROUTINE COLD_START_EXISTING_FILE: NOVALUE=
;  1033          2067  1
;  1034          2068  1 !++
;  1035          2069  1 !
;  1036          2070  1 ! FUNCTIONAL DESCRIPTION:
;  1037          2071  1 !       This routine reinitializes an existing queue file.
;  1038          2072  1 !
;  1039          2073  1 ! INPUT PARAMETERS:
;  1040          2074  1 !       NONE
;  1041          2075  1 !
;  1042          2076  1 ! IMPLICIT INPUTS:
;  1043          2077  1 !       NONE
;  1044          2078  1 !
;  1045          2079  1 ! OUTPUT PARAMETERS:
;  1046          2080  1 !       NONE
;  1047          2081  1 !
;  1048          2082  1 ! IMPLICIT OUTPUTS:
;  1049          2083  1 !       NONE
;  1050          2084  1 !
;  1051          2085  1 ! ROUTINE VALUE:
;  1052          2086  1 !       NONE
;  1053          2087  1 !
;  1054          2088  1 ! SIDE EFFECTS:
;  1055          2089  1 !       NONE
;  1056          2090  1 !
;  1057          2091  1 !--
;  1058          2092  1
;  1059          2093  2 BEGIN
;  1060          2094  2 LOCAL
;  1061          2095  2         HIGHEST_ENTRY_NUMBER,           ! Expected highest entry number
;  1062          2096  2         HIGHEST_ENTRY_OFFSET,           ! Highest nonzero bitmap slot
;  1063          2097  2         FREE_N,
;  1064          2098  2         BUFFER:                 BBLOCK[SYM$S_SYM],
;  1065          2099  2         BITMAP_RETADR:          VECTOR[2],
;  1066          2100  2         SAVED_PENDING_BATCH_LIST,
;  1067          2101  2         SAVED_PENDING_PRINT_LIST,
;  1068          2102  2         SAVED_TIMER_LIST,
;  1069          2103  2         STATUS;
;  1070          2104  2 GLOBAL REGISTER
;  1071          2105  2         RTM = 10:               REF VECTOR[,BYTE,SIGNED],
;  1072          2106  2         SQH = 11:               REF BBLOCK;
;  1073          2107  2
;  1074          2108  2
;  1075          2109  2 ! Allocate and initialize the record type map.  This vector records the
;  1076          2110  2 ! record type of each record encountered.  It is initially -1, so that
;  1077          2111  2 ! multiply allocated and lost records can be readily distinguished.
;  1078          2112  2 !
;  1079        P 2113  2 STATUS = $EXPREG(
;  1080        P 2114  2     PAGCNT=(.QUEUE_FAB[FAB$L_ALQ] + 511) ^ -9,
;  1081          2115  2     RETADR=BITMAP_RETADR);
;  1082          2116  2 IF NOT .STATUS THEN SIGNAL(JBC$_ALLOCMEM OR STS$K_SEVERE, 0, .STATUS);
;  1083          2117  2 INCRA P FROM .BITMAP_RETADR[0] TO .BITMAP_RETADR[1] AND NOT 511 BY 512 DO
;  1084          2118  2     CH$FILL(-1, 512, .P);
;  1085          2119  2
;  1086          2120  2
;  1087          2121  2 ! Read the queue header and initialize global registers for the TREE routine.
;  1088          2122  2 !
```

```
;  1089          2123  2 RTM = .BITMAP_RETADR[0];
;  1090          2124  2 SQH = READ_RECORD(SQH$K_RECNO);
;  1091          2125  2
;  1092          2126  2
;  1093          2127  2 ! Process a discrepancy between the queue file allocation and the highest
;  1094          2128  2 ! record number recorded in the queue file.
;  1095          2129  2 !
;  1096          2130  2 IF .SQH[SQH$L_HIGHEST_RECORD] NEQ .QUEUE_FAB[FAB$L_ALQ]
;  1097          2131  2 THEN
;  1098          2132  3     BEGIN
;  1099          2133  3
;  1100          2134  3     ! If the highest record number is beyond the allocated space, it is
;  1101          2135  3     ! an error.
;  1102          2136  3     !
;  1103          2137  3     IF .SQH[SQH$L_HIGHEST_RECORD] GTRU .QUEUE_FAB[FAB$L_ALQ]
;  1104          2138  3     THEN
;  1105          2139  3         SQH[SQH$L_HIGHEST_RECORD] = .QUEUE_FAB[FAB$L_ALQ];
;  1106          2140  3
;  1107          2141  3
;  1108          2142  3     ! Presumably a crash occurred while the file was being extended. Ensure that
;  1109          2143  3     ! records beyond the highest recorded record number are initialized and
;  1110          2144  3     ! process them later as lost records.
;  1111          2145  3     !
;  1112          2146  3     CH$FILL(0, SYM$S_SYM, BUFFER);
;  1113          2147  3     QUEUE_RAB[RAB$L_RBF] = BUFFER;
;  1114          2148  3     IF .FLAGS[FLAGS_V_QUEUE_SHARED]
;  1115          2149  3         THEN QUEUE_RAB[RAB$L_ROP] = RAB$M_UIF OR RAB$M_NLK
;  1116          2150  3         ELSE QUEUE_RAB[RAB$L_ROP] = RAB$M_UIF;
;  1117          2151  3     DECR REC_N FROM .QUEUE_FAB[FAB$L_ALQ] TO .SQH[SQH$L_HIGHEST_RECORD] + 1 DO
;  1118          2152  4         BEGIN
;  1119          2153  4         QUEUE_RAB[RAB$L_KBF] = REC_N;
;  1120          2154  4         DIAG_COUNT[1] = .DIAG_COUNT[1] + 1;
;  1121          2155  5         IF NOT $PUT(RAB=QUEUE_RAB)
;  1122          2156  4         THEN
;  1123          2157  4             SIGNAL_FILE_ERROR(
;  1124          2158  4                 JBC$_WRITEERR + STS$K_SEVERE,
;  1125          2159  4                 QUEUE_FAB, QUEUE_RAB);
;  1126          2160  3         END;
;  1127          2161  3     SQH[SQH$L_HIGHEST_RECORD] = .QUEUE_FAB[FAB$L_ALQ];
;  1128          2162  2     END;
;  1129          2163  2
;  1130          2164  2
;  1131          2165  2 ! Traverse the free list.  This action must precede the deletion of any
;  1132          2166  2 ! other records.
;  1133          2167  2 !
;  1134          2168  2 DIAG_TRACE[8] = 1;                          ! ***** diagnostic info *****
;  1135          2169  2 TREE7.SQH[SQH$L_FREE_LIST], 0, FALSE);
;  1136          2170  2
;  1137          2171  2
;  1138          2172  2 ! Traverse the characteristic definition list.
;  1139          2173  2 !
;  1140          2174  2 DIAG_TRACE[8] = 2;                          ! ***** diagnostic info *****
;  1141          2175  2 TREE7.SQH[SQH$L_CHARACTERISTIC_LIST], SYM$K_SCX, FALSE);
;  1142          2176  2
;  1143          2177  2
;  1144          2178  2 ! Compute the expected value of the highest entry number based on the
;  1145          2179  2 ! existing extension bitmap records, and correct the value recorded in the
```

```
;  1146     2180  2 ! queue header.
;  1147     2181  2 !
;  1148     2182  2 DIAG_TRACE[8] = 3;                          ! ***** diagnostic info *****
;  1149     2183  2 HIGHEST_ENTRY_NUMBER =
;  1150     2184  2     SQH$S_ENTRY_BITMAP * 8 +
;  1151     2185  2     (SYM$S_DATA * 8) * (SQH$S_ENTRY_BITMAP_VECTOR/4);
;  1152     2186  2 HIGHEST_ENTRY_OFFSET = -1;
;  1153     2187  2 DECR N FROM SQH$S_ENTRY_BITMAP_VECTOR/4-1 TO 0 DO
;  1154     2188  3     BEGIN
;  1155     2189  3     IF .VECTOR[SQH[SQH$L_ENTRY_BITMAP_VECTOR], .N] NEQ 0
;  1156     2190  3     THEN
;  1157     2191  4         BEGIN
;  1158     2192  4         HIGHEST_ENTRY_OFFSET = .N;
;  1159     2193  4         EXITLOOP;
;  1160     2194  3         END;
;  1161     2195  3     HIGHEST_ENTRY_NUMBER = .HIGHEST_ENTRY_NUMBER - (SYM$S_DATA * 8);
;  1162     2196  2     END;
;  1163     2197  2 SQH[SQH$L_HIGHEST_ENTRY_NUMBER] = .HIGHEST_ENTRY_NUMBER;
;  1164     2198  2
;  1165     2199  2
;  1166     2200  2 ! Correct the next entry number value if necessary.
;  1167     2201  2 !
;  1168     2202  2 IF .SQH[SQH$L_NEXT_ENTRY_NUMBER] EQLU 0
;  1169     2203  2 OR .SQH[SQH$L_NEXT_ENTRY_NUMBER] GTRU .HIGHEST_ENTRY_NUMBER + 1
;  1170     2204  2 THEN
;  1171     2205  2     SQH[SQH$L_NEXT_ENTRY_NUMBER] = 1;
;  1172     2206  2
;  1173     2207  2
;  1174     2208  2 ! Traverse and reinitialize the extension bitmap records, ensuring that each one
;  1175     2209  2 ! exists.  This must occur before any job lists are scanned.
;  1176     2210  2 !
;  1177     2211  2 INCR N FROM 0 TO .HIGHEST_ENTRY_OFFSET DO
;  1178     2212  3     BEGIN
;  1179     2213  3     LOCAL
;  1180     2214  3         SEB_N;                     ! Record number of extension bitmap
;  1181     2215  3
;  1182     2216  3     SEB_N = .VECTOR[SQH[SQH$L_ENTRY_BITMAP_VECTOR], .N];
;  1183     2217  3     IF .SEB_N EQL 0 THEN SIGNAL(JBC$_QUEFORMAT);
;  1184     2218  3     TREE(.SEB_N, SYM$K_ENTRY_BITMAP, FALSE);
;  1185     2219  2     END;
;  1186     2220  2 CH$FILL(0, SQH$S_ENTRY_BITMAP, SQH[SQH$B_ENTRY_BITMAP]);
;  1187     2221  2
;  1188     2222  2
;  1189     2223  2 ! Traverse the form index list.  This must occur before any job lists are
;  1190     2224  2 ! scanned and before the queue list second pass.
;  1191     2225  2 !
;  1192     2226  2 DIAG_TRACE[8] = 4;                          ! ***** diagnostic info *****
;  1193     2227  2 TREE(.SQH[SQH$L_FORM_INDEX_LIST], SYM$K_SFX, FALSE);
;  1194     2228  2
;  1195     2229  2
;  1196     2230  2 ! Traverse and delete the incomplete service list.
;  1197     2231  2 !
;  1198     2232  2 DIAG_TRACE[8] = 5;                          ! ***** diagnostic info *****
;  1199     2233  2 TREE(.SQH[SQH$L_INCOMPLETE_SERVICE_LIST], SYM$K_SRQ, TRUE);
;  1200     2234  2 SQH[SQH$L_INCOMPLETE_SERVICE_LIST] = 0;
;  1201     2235  2
;  1202     2236  2
```

```
;  1203          2237  2 ! Traverse the queue index list.  This must occur before any job lists are
;  1204          2238  2 ! scanned.
;  1205          2239  2 !
;  1206          2240  2 DIAG_TRACE[8] = 6;                          ! ***** diagnostic info *****
;  1207          2241  2 TREET.SQH[SQH$L_QUEUE_INDEX_LIST], SYM$K_SQX, FALSE);
;  1208          2242  2
;  1209          2243  2
;  1210          2244  2 ! Traverse and delete the open job list.
;  1211          2245  2 !
;  1212          2246  2 DIAG_TRACE[8] = 7;                          ! ***** diagnostic info *****
;  1213          2247  2 TREET.SQH[SQH$L_OPEN_LIST], SYM$K_SJH, TRUE);
;  1214          2248  2 SQH[SQH$L_OPEN_LIST] = 0;
;  1215          2249  2 SQH[SQH$L_OPEN_LIST_END] = 0;
;  1216          2250  2
;  1217          2251  2
;  1218          2252  2 ! Save and reinitialize the pending and timer lists.  This must occur before
;  1219          2253  2 ! any job lists are scanned.
;  1220          2254  2 !
;  1221          2255  2 SAVED_PENDING_BATCH_LIST = .SQH[SQH$L_PENDING_BATCH_LIST];
;  1222          2256  2 SAVED_PENDING_PRINT_LIST = .SQH[SQH$L_PENDING_PRINT_LIST];
;  1223          2257  2 SAVED_TIMER_LIST = .SQH[SQH$L_TIMER_LIST];
;  1224          2258  2 SQH[SQH$L_PENDING_BATCH_LIST] = 0;
;  1225          2259  2 SQH[SQH$L_PENDING_BATCH_LIST_END] = 0;
;  1226          2260  2 SQH[SQH$L_PENDING_PRINT_LIST] = 0;
;  1227          2261  2 SQH[SQH$L_PENDING_PRINT_LIST_END] = 0;
;  1228          2262  2 SQH[SQH$L_TIMER_LIST] = 0;
;  1229          2263  2 SQH[SQH$L_TIMER_LIST_END] = 0;
;  1230          2264  2
;  1231          2265  2
;  1232          2266  2 ! Traverse the queue index list (pass 2).
;  1233          2267  2 !
;  1234          2268  2 DIAG_TRACE[8] = 8;                          ! ***** diagnostic info *****
;  1235          2269  2 TREET.SQH[SQH$L_QUEUE_INDEX_LIST], SYM$K_SQX OR M_SMQ_P2 OR M_NO_MARK, FALSE);
;  1236          2270  2
;  1237          2271  2 ! Find any deallocated job records that remain in the CURRENT_LIST of any
;  1238          2272  2 ! queues.
;  1239          2273  2 !
;  1240          2274  2 DIAG_TRACE[8] = 9;                          ! ***** diagnostic info *****
;  1241          2275  2 COLD_START_CURRENT_LIST();
;  1242          2276  2
;  1243          2277  2 ! Traverse the queue index list (pass 3).
;  1244          2278  2 !
;  1245          2279  2 DIAG_TRACE[8] = 10;                         ! ***** diagnostic info *****
;  1246          2280  2 TREET.SQH[SQH$L_QUEUE_INDEX_LIST], SYM$K_SQX OR M_SMQ_P3 OR M_NO_MARK, FALSE);
;  1247          2281  2
;  1248          2282  2
;  1249          2283  2 ! Traverse the pending lists and the timer list.
;  1250          2284  2 !
;  1251          2285  2 DIAG_TRACE[8] = 11;                         ! ***** diagnostic info *****
;  1252          2286  2 TREET.SAVED_PENDING_BATCH_LIST, SYM$K_SJH, FALSE);
;  1253          2287  2 DIAG_TRACE[8] = 12;                         ! ***** diagnostic info *****
;  1254          2288  2 TREET.SAVED_PENDING_PRINT_LIST, SYM$K_SJH, FALSE);
;  1255          2289  2 DIAG_TRACE[8] = 13;                         ! ***** diagnostic info *****
;  1256          2290  2 TREET.SAVED_TIMER_LIST, SYM$K_SJH, FALSE);
;  1257          2291  2
;  1258          2292  2
;  1259          2293  2 ! Pass over the lost blocks looking for lost job headers.
```

```
;  1260          2294  2 !
;  1261          2295  2 DIAG_TRACE[8] = 14;                    ! ***** diagnostic info *****
;  1262          2296  2 INCR REC_N FROM 2 TO .QUEUE_FAB[FAB$L_ALQ] DO
;  1263          2297  3     BEGIN
;  1264          2298  3     IF .RTM[.REC_N-1] LSS 0
;  1265          2299  3     THEN
;  1266          2300  4         BEGIN
;  1267          2301  4         LOCAL
;  1268          2302  4             REC:        REF BBLOCK;     ! Pointer to lost record
;  1269          2303  4
;  1270          2304  4         REC = READ_RECORD(.REC_N);
;  1271          2305  4         IF .REC[SYM$B_TYPE] EQL SYM$K_SJH
;  1272          2306  4             THEN TREE(.REC_N, SYM$K_SJH OR M_NO_TRAVERSE, FALSE);
;  1273          2307  4         RELEASE_RECORD(.REC_N);
;  1274          2308  3         END;
;  1275          2309  2     END;
;  1276          2310  2
;  1277          2311  2 ! Indicate last pass is complete.
;  1278          2312  2 !
;  1279          2313  2 DIAG_TRACE[8] = -65536;                 ! ***** diagnostic info *****
;  1280          2314  2
;  1281          2315  2 ! Link all remaining lost blocks to the free list.
;  1282          2316  2 !
;  1283          2317  2 FREE_N = .SQH[SQH$L_FREE_LIST];
;  1284          2318  2 CH$FILL(0, SYM$S_SYM, BUFFER);
;  1285          2319  2 QUEUE_RAB[RAB$L_RBF] = BUFFER;
;  1286          2320  2 IF .FLAGS[FLAGS_V_QUEUE_SHARED]
;  1287          2321  2     THEN QUEUE_RAB[RAB$L_ROP] = RAB$M_UIF OR RAB$M_NLK
;  1288          2322  2     ELSE QUEUE_RAB[RAB$L_ROP] = RAB$M_UIF;
;  1289          2323  2 DECR REC_N FROM .QUEUE_FAB[FAB$L_ALQ] TO 2 DO
;  1290          2324  3     BEGIN
;  1291          2325  3     IF .RTM[.REC_N-1] LSS 0
;  1292          2326  3     THEN
;  1293          2327  4         BEGIN
;  1294          2328  4         BUFFER[SYM$L_LINK] = .FREE_N;
;  1295          2329  4         FREE_N = .REC_N;
;  1296          2330  4         QUEUE_RAB[RAB$L_KBF] = REC_N;
;  1297          2331  4         DIAG_TRACE[10] = .DIAG_TRACE[10] + 1;    ! Count lost blocks recovered
;  1298          2332  4         DIAG_COUNT[1] = .DIAG_COUNT[1] + 1;
;  1299          2333  5         IF NOT $PUT(RAB=QUEUE_RAB)
;  1300          2334  4         THEN
;  1301          2335  4             SIGNAL_FILE_ERROR(
;  1302          2336  4                 JBC$_WRITEERR + STS$K_SEVERE,
;  1303          2337  4                 QUEUE_FAB, QUEUE_RAB);
;  1304          2338  3         END;
;  1305          2339  2     END;
;  1306          2340  2 SQH[SQH$L_FREE_LIST] = .FREE_N;
;  1307          2341  2 SQH[SQH$L_HIGHEST_RECORD] = .QUEUE_FAB[FAB$L_ALQ];
;  1308          2342  2
;  1309          2343  2
;  1310          2344  2 ! Ensure that the end of file pointer is correct.
;  1311          2345  2 !
;  1312          2346  3 IF NOT $FLUSH(RAB=QUEUE_RAB)
;  1313          2347  2 THEN
;  1314          2348  2     SIGNAL_FILE_ERROR(
;  1315          2349  2         JBC$_WRITEERR + STS$K_SEVERE,
;  1316          2350  2         QUEUE_FAB, QUEUE_RAB);
```

```
;  1317          2351  2
;  1318          2352  2
;  1319          2353  2 ! If the timer list is not empty, set a timer on the first job.
;  1320          2354  2 !
;  1321          2355  2 IF .SQH[SQH$L_TIMER_LIST] NEQ 0
;  1322          2356  2 THEN
;  1323          2357  3     BEGIN
;  1324          2358  3     LOCAL
;  1325          2359  3         SJH_N,                          ! Record number of SJH
;  1326          2360  3         SJH:            REF BBLOCK,     ! Pointer to SJH
;  1327          2361  3         STATUS;                         ! Status return
;  1328          2362  3
;  1329          2363  3     SJH = READ_RECORD(SJH_N = .SQH[SQH$L_TIMER_LIS ]);
;  1330        P 2364  3     STATUS = $SETIMR(
;  1331        P 2365  3         DAYTIM=SJH[SJH$Q_AFTER_TIME],
;  1332        P 2366  3         ASTADR=AFTER_AST,
;  1333          2367  3         REQIDT=JBC$K_AFTER_IDT);
;  1334          2368  3     IF NOT .STATUS
;  1335          2369  3     THEN
;  1336          2370  3         SIGNAL(JBC$_SETIMR OR STS$K_ERROR, 0, .STATUS);
;  1337          2371  3     RELEASE_RECORD(.SJH_N);
;  1338          2372  2     END;
;  1339          2373  2
;  1340          2374  2
;  1341          2375  2 ! Rewrite the queue header.
;  1342          2376  2 !
;  1343          2377  2 REWRITE_RECORD(SQH$K_RECNO);
;  1344          2378  2
;  1345          2379  2
;  1346          2380  2 ! Release the record type bitmap to the free memory list.
;  1347          2381  2 !
;  1348          2382  2 DECRA P FROM .BITMAP_RETADR[1] AND NOT 511 TO .BITMAP_RETADR[0] BY 512 DO
;  1349          2383  2     DEALLOCATE_MEMORY(.P);
;  1350          2384  1 END;

                                                .EXTRN  SYS$EXPREG, SYS$PUT
                                                .EXTRN  SYS$FLUSH, SYS$SETIMR

                                  0FFC 00000 COLD_START_EXISTING_FILE:
                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11    ; 2066
                          59      FB46  CF  9E 00002    MOVAB   TREE, R9                        ;
                          58 00000000'  EF  9E 00007    MOVAB   DIAG_TRACE+32, R8               ;
                          5E      FDF0  CE  9E 0000E    MOVAB   -528(SP), SP                    ;
                                        7E  7C 00013    CLRQ    -(SP)                           ; 2115
                                    10  AE  9F 00015    PUSHAB  BITMAP_RETADR                   ;
                50      0304  C8 000001FF 8F  C1 00018    ADDL3   #511, QUEUE_FAB+16, R0      ;
                7E            50        F7  8F  78 00022    ASHL    #-9, R0, -(SP)              ;
                   00000000G  00            04  FB 00027    CALLS   #4, SYS$EXPREG              ;
                              11            50  E8 0002E    BLBS    STATUS, 1$                  ; 2116
                                            50  DD 00031    PUSHL   STATUS                      ;
                                            7E  D4 00033    CLRL    -(SP)                       ;
                                 0004840C   8F  DD 00035    PUSHL   #295948                     ;
                   00000000G  00            03  FB 0003B    CALLS   #3, LIB$SIGNAL              ;
                57        0C  AE 000001FF   8F  CB 00042 1$:  BICL3   #511, BITMAP_RETADR+4, R7 ; 2117
                              5A        08  AE  D0 0004B    MOVL    BITMAP_RETADR, P            ;
```

```
                                       0E  11 0004F          BRB     3$                                     ;
 0200   8F     FF   8F            6E   00  2C 00051  2$:     MOVC5   #0, (SP), #-1, #512, (P)               ; 2118
                                       6A     00059                                                         ;
                            5A   0200  CA  9E 0005A          MOVAB   512(R10), P                            ;
                            57         5A  D1 0005F  3$:     CMPL    P, R7                                  ;
                                       ED  1B 00062          BLEQU   2$                                     ;
                            5A     08  AE  D0 00064          MOVL    BITMAP_RETADR, RTM                     ; 2123
                                       01  DD 00068          PUSHL   #1                                     ; 2124
                 00000000G  EF         01  FB 0006A          CALLS   #1, READ_RECORD                        ;
                            5B         50  D0 00071          MOVL    R0, SQH                                ;
                            56   0304  C8  D0 00074          MOVL    QUEUE_FAB+16, R6                       ; 2130
                            56     40  AB  D1 00079          CMPL    64(SQH), R6                            ;
                                       6D  13 0007D          BEQL    10$                                    ;
                                       04  1B 0007F          BLEQU   4$                                     ; 2137
                       40   AB         56  D0 00081          MOVL    R6, 64(SQH)                            ; 2139
 0200   8F          00            6E   00  2C 00085  4$:     MOVC5   #0, (SP), #0, #512, BUFFER             ; 2146
                                   10  AE     0008C                                                         ;
                     036C   C8     10  AE  9E 0008E          MOVAB   BUFFER, QUEUE_RAB+40                   ; 2147
               0B    01E0   C8         03  E1 00094          BBC     #3, FLAGS, 5$                          ; 2148
                     0348   C8 00100010 8F D0 0009A          MOVL    #1048592, QUEUE_RAB+4                  ; 2149
                                       05  11 000A3          BRB     6$                                     ;
                     0348   C8         10  D0 000A5  5$:     MOVL    #16, QUEUE_RAB+4                       ; 2150
               52      40   AB         01  C1 000AA  6$:     ADDL3   #1, 64(SQH), R2                        ; 2151
                            6E         56  D0 000AF          MOVL    R6, REC_N                              ;
                                       2D  11 000B2          BRB     9$                                     ;
                     0374   C8         6E  9E 000B4  7$:     MOVAB   REC_N, QUEUE_RAB+48                    ; 2153
                                   44  A8  D6 000B9          INCL    DIAG_COUNT+4                           ; 2154
                                 0344  C8  9F 000BC          PUSHAB  QUEUE_RAB                              ; 2155
                 00000000G  00         01  FB 000C0          CALLS   #1, SYS$PUT                            ;
                            15         50  E8 000C7          BLBS    R0, 8$                                 ;
                                 0344  C8  9F 000CA          PUSHAB  QUEUE_RAB                              ; 2157
                                 02F4  C8  9F 000CE          PUSHAB  QUEUE_FAB                              ;
                             000410D4  8F  DD 000D2          PUSHL   #266452                                ; 2158
                 00000000G  EF         03  FB 000D8          CALLS   #3, SIGNAL_FILE_ERROR                  ;
                                       6E  D7 000DF  8$:     DECL    REC_N                                  ; 2151
                            52         6E  D1 000E1  9$:     CMPL    REC_N, R2                              ;
                                       CE  18 000E4          BGEQ    7$                                     ;
                       40   AB   0304  C8  D0 000E6          MOVL    QUEUE_FAB+16, 64(SQH)                  ; 2161
                            68         01  D0 000EC  10$:    MOVL    #1, DIAG_TRACE+32                      ; 2168
                                       7E  7C 000EF          CLRQ    -(SP)                                  ; 2169
                                   38  AB  DD 000F1          PUSHL   56(SQH)                                ;
                            69         03  FB 000F4          CALLS   #3, TREE                               ;
                            68         02  D0 000F7          MOVL    #2, DIAG_TRACE+32                      ; 2174
                            7E         01  7D 000FA          MOVQ    #1, -(SP)                              ; 2175
                                   10  AB  DD 000FD          PUSHL   16(SQH)                                ;
                            69         03  FB 00100          CALLS   #3, TREE                               ;
                            68         03  D0 00103          MOVL    #3, DIAG_TRACE+32                      ; 2182
                            50   8500  8F  3C 00106          MOVZWL  #34048, HIGHEST_ENTRY_NUMBER           ; 2184
                            54         01  CE 0010B          MNEGL   #1, HIGHEST_ENTRY_OFFSET               ; 2186
                            52     14  AB  9E 0010E          MOVAB   20(SQH), R2                            ; 2189
                            51         07  D0 00112          MOVL    #7, N                                  ;
                                     6241  D5 00115  11$:    TSTL    (R2)[N]                                ;
                                       05  13 00118          BEQL    12$                                    ;
                            54         51  D0 0011A          MOVL    N, HIGHEST_ENTRY_OFFSET                ; 2192
                                       08  11 0011D          BRB     13$                                    ; 2191
                            50   F060  C0  9E 0011F  12$:    MOVAB   -4000(R0), HIGHEST_ENTRY_NUMBER        ; 2195
                            EE         51  F4 00124          SOBGEQ  N, 11$                                 ; 2187
```

```
                                3C    AB                50    D0 00127 13$:    MOVL    HIGHEST_ENTRY_NUMBER, 60(SQH)                ; 2197
                                            48    AB    D5 0012B               TSTL    72(SQH)                                      ; 2202
                                                  08    13 0012E               BEQL    14$                                          ;
                                                  50    D6 00130               INCL    R0                                           ; 2203
                                      50          48    AB    D1 00132         CMPL    72(SQH), R0                                  ;
                                                  04    1B 00136               BLEQU   15$                                          ;
                                48    AB                01    D0 00138 14$:    MOVL    #1, 72(SQH)                                  ; 2205
                                      52                01    CE 0013C 15$:    MNEGL   #1, N                                        ; 2211
                                                        1C    11 0013F         BRB     18$                                          ;
                                      53          14 AB42    D0 00141 16$:    MOVL    20(SQH)[N], SEB_N                            ; 2216
                                                        0D    12 00146         BNEQ    17$                                          ; 2217
                                            00048440    8F    DD 00148         PUSHL   #296000                                      ;
                          00000000G   00                01    FB 0014E         CALLS   #1, LIB$SIGNAL                               ;
                                      7E                0A    7D 00155 17$:    MOVQ    #10, -(SP)                                   ; 2218
                                                        53    DD 00158         PUSHL   SEB_N                                        ;
                                      69                03    FB 0015A         CALLS   #3, TREE                                     ;
                          E0          52                54    F3 0015D 18$:    AOBLEQ  HIGHEST_ENTRY_OFFSET, N, 16$                 ; 2211
    0100  8F              00          6E                00    2C 00161         MOVC5   #0, (SP), #0, #256, 256(SQH)                 ; 2220
                                                  0100  CB    00168                                                                 ;
                                      68                04    D0 0016B         MOVL    #4, DIAG_TRACE+32                            ; 2226
                                      7E                03    7D 0016E         MOVQ    #3, -(SP)                                    ; 2227
                                                  34    AB    DD 00171         PUSHL   52(SQH)                                      ;
                                      69                03    FB 00174         CALLS   #3, TREE                                     ;
                                      68                05    D0 00177         MOVL    #5, DIAG_TRACE+32                            ; 2232
                                                        01    DD 0017A         PUSHL   #1                                           ; 2233
                                                        09    DD 0017C         PUSHL   #9                                           ;
                                                  44    AB    DD 0017E         PUSHL   68(SQH)                                      ;
                                      69                03    FB 00181         CALLS   #3, TREE                                     ;
                                                  44    AB    D4 00184         CLRL    68(SQH)                                      ; 2234
                                      68                06    D0 00187         MOVL    #6, DIAG_TRACE+32                            ; 2240
                                      7E                05    7D 0018A         MOVQ    #5, -(SP)                                    ; 2241
                                                  64    AB    DD 0018D         PUSHL   100(SQH)                                     ;
                                      69                03    FB 00190         CALLS   #3, TREE                                     ;
                                      68                07    D0 00193         MOVL    #7, DIAG_TRACE+32                            ; 2246
                                                        01    DD 00196         PUSHL   #1                                           ; 2247
                                                        07    DD 00198         PUSHL   #7                                           ;
                                                  4C    AB    DD 0019A         PUSHL   76(SQH)                                      ;
                                      69                03    FB 0019D         CALLS   #3, TREE                                     ;
                                                  4C    AB    7C 001A0         CLRQ    76(SQH)                                      ; 2248
                                      54          54    AB    D0 001A3         MOVL    84(SQH), SAVED_PENDING_BATCH_LIST            ; 2255
                                      53          5C    AB    D0 001A7         MOVL    92(SQH), SAVED_PENDING_PRINT_LIST            ; 2256
                                      52          68    AB    D0 001AB         MOVL    104(SQH), SAVED_TIMER_LIST                   ; 2257
                                                  54    AB    7C 001AF         CLRQ    84(SQH)                                      ; 2258
                                                  5C    AB    7C 001B2         CLRQ    92(SQH)                                      ; 2260
                                                  68    AB    7C 001B5         CLRQ    104(SQH)                                     ; 2262
                                      68                08    D0 001B8         MOVL    #8, DIAG_TRACE+32                            ; 2268
                                                        7E    D4 001BB         CLRL    -(SP)                                        ; 2269
                                      7E          0905  8F    3C 001BD         MOVZWL  #2309, -(SP)                                 ;
                                                  64    AB    DD 001C2         PUSHL   100(SQH)                                     ;
                                      69                03    FB 001C5         CALLS   #3, TREE                                     ;
                                      68                09    D0 001C8         MOVL    #9, DIAG_TRACE+32                            ; 2274
    0400  C9                                      00    FB 001CB         CALLS   #0, COLD_START_CURRENT_LIST                  ; 2275
                                      68                0A    D0 001D0         MOVL    #10, DIAG_TRACE+32                           ; 2279
                                                        7E    D4 001D3         CLRL    -(SP)                                        ; 2280
                                      7E          1105  8F    3C 001D5         MOVZWL  #4357, -(SP)                                 ;
                                                  64    AB    DD 001DA         PUSHL   100(SQH)                                     ;
                                      69                03    FB 001DD         CALLS   #3, TREE                                     ;
```

```
                                68            0B  DO 001E0          MOVL    #11, DIAG_TRACE+32                      ; 2285
                                7E            07  7D 001E3          MOVQ    #7, -(SP)                               ; 2286
                                              54  DD 001E6          PUSHL   SAVED_PENDING_BATCH_LIST                ;
                                69            03  FB 001E8          CALLS   #3, TREE                                ;
                                68            0C  DO 001EB          MOVL    #12, DIAG_TRACE+32                      ; 2287
                                7E            07  7D 001EE          MOVQ    #7, -(SP)                               ; 2288
                                              53  DD 001F1          PUSHL   SAVED_PENDING_PRINT_LIST                ;
                                69            03  FB 001F3          CALLS   #3, TREE                                ;
                                68            0D  DO 001F6          MOVL    #13, DIAG_TRACE+32                      ; 2289
                                7E            07  7D 001F9          MOVQ    #7, -(SP)                               ; 2290
                                              52  DD 001FC          PUSHL   SAVED_TIMER_LIST                        ;
                                69            03  FB 001FE          CALLS   #3, TREE                                ;
                                68            0E  DO 00201          MOVL    #14, DIAG_TRACE+32                      ; 2295
                                53      0304  C8  DO 00204          MOVL    QUEUE_FAB+16, R3                        ; 2296
                                52            01  DO 00209          MOVL    #1, REC_N                               ;
                                              2A  11 0020C          BRB     21$                                     ;
                                   FF A24A    95  00020E 19$:       TSTB    -1(REC_N)[RTM]                          ; 2298
                                              24  18 00212          BGEQ    21$                                     ;
                                              52  DD 00214          PUSHL   REC_N                                   ; 2304
                      00000000G EF            01  FB 00216          CALLS   #1, READ_RECORD                         ;
                                07      04    A0  91 00219          CMPB    4(REC), #7                              ; 2305
                                              0C  12 00221          BNEQ    20$                                     ;
                                              7E  D4 00223          CLRL    -(SP)                                   ; 2306
                                7E      0407  8F  3C 00225          MOVZWL  #1031, -(SP)                            ;
                                              52  DD 0022A          PUSHL   REC_N                                   ;
                                69            03  FB 0022C          CALLS   #3, TREE                                ;
                                              52  DD 0022F 20$:     PUSHL   REC_N                                   ; 2307
                      00000000G EF            01  FB 00231          CALLS   #1, RELEASE_RECORD                      ;
                             D2 52            53  F3 00238 21$:     AOBLEQ  R3, REC_N, 19$                          ; 2296
                                68 FFFF0000  8F  DO 0023C          MOVL    #-65536, DIAG_TRACE+32                  ; 2313
                                56      38    AB  DO 00243          MOVL    56(SQH), FREE_N                         ; 2317
   0200  8F                   00            6E  00 2C 00247          MOVC5   #0, (SP), #0, #512, BUFFER              ; 2318
                                                10  AE    0024E                                                    ;
                          036C  C8        10    AE  9E 00250          MOVAB   BUFFER, QUEUE_RAB+40                    ; 2319
                       0B 01E0  C8              03  E1 00256          BBC     #3, FLAGS, 22$                          ; 2320
                          0348  C8  00100010    8F  DO 0025C          MOVL    #1048592, QUEUE_RAB+4                   ; 2321
                                              05  11 00265          BRB     23$                                     ;
                          0348  C8            10  DO 00267 22$:     MOVL    #16, QUEUE_RAB+4                        ; 2322
                            04  AE      0304  C8  DO 0026C 23$:     MOVL    QUEUE_FAB+16, REC_N                     ; 2325
                                              44  11 00272          BRB     26$                                     ;
                       50             5A      04  AE  C1 00274 24$:     ADDL3   REC_N, RTM, R0                          ;
                                         FF    A0  95 00279          TSTB    -1(R0)                                  ;
                                              37  18 0027C          BGEQ    25$                                     ;
                             10  AE            56  DO 0027E          MOVL    FREE_N, BUFFER                          ; 2328
                                56      04    AE  DO 00282          MOVL    REC_N, FREE_N                           ; 2329
                          0374  C8      04    AE  9E 00286          MOVAB   REC_N, QUEUE_RAB+48                     ; 2330
                                        08    AB  D6 0028C          INCL    DIAG_TRACE+40                           ; 2331
                                        44    AB  D6 0028F          INCL    DIAG_COUNT+4                            ; 2332
                                      0344    C8  9F 00292          PUSHAB  QUEUE_RAB                               ; 2333
                      00000000G 00            01  FB 00296          CALLS   #1, SYS$PUT                             ;
                                15            50  E8 0029D          BLBS    R0, 25$                                 ;
                                      0344    C8  9F 002A0          PUSHAB  QUEUE_RAB                               ; 2335
                                      02F4    C8  9F 002A4          PUSHAB  QUEUE_FAB                               ;
                                   000410D4   8F  DD 002A8          PUSHL   #266452                                 ; 2336
                      00000000G EF            03  FB 002AE          CALLS   #3, SIGNAL_FILE_ERROR                   ;
                                        04    AE  D7 002B5 25$:     DECL    REC_N                                   ; 2323
                                02      04    AE  D1 002B8 26$:     CMPL    REC_N, #2                               ;
```

```
                                    B6  18 002BC         BGEQ    24$
             38   AB                56  D0 002BE         MOVL    FREE_N, 56(SQH)                          ; 2340
             40   AB       0304     C8  D0 002C2         MOVL    QUEUE_FAB+16, 64(SQH)                    ; 2341
                           0344     C8  9F 002C8         PUSHAB  QUEUE_RAB                                ; 2346
       00000000G  00                01  FB 002CC         CALLS   #1, SYS$FLUSH
                  15                50  E8 002D3         BLBS    R0, 27$
                           0344     C8  9F 002D6         PUSHAB  QUEUE_RAB                                ; 2348
                           02F4     C8  9F 002DA         PUSHAB  QUEUE_FAB
                       000410D4     8F  DD 002DE         PUSHL   #266452                                  ; 2349
       00000000G  EF                03  FB 002E4         CALLS   #3, SIGNAL_FILE_ERROR
                             68     AB  D5 002EB 27$:    TSTL    104(SQH)                                 ; 2355
                                    3F  13 002EE         BEQL    29$
                  52         68     AB  D0 002F0         MOVL    104(SQH), SJH_N                          ; 2363
                                    52  DD 002F4         PUSHL   SJH_N
       00000000G  EF                01  FB 002F6         CALLS   #1, READ_RECORD
                                    01  DD 002FD         PUSHL   #1                                       ; 2367
                      00000000G     EF  9F 002FF         PUSHAB  AFTER_AST
                           0098     C0  9F 00305         PUSHAB  152(SJH)
                                    7E  D4 00309         CLRL    -(SP)
       00000000G  00                04  FB 0030B         CALLS   #4, SYS$SETIMR
                  11                50  E8 00312         BLBS    STATUS, 28$                              ; 2368
                                    50  DD 00315         PUSHL   STATUS                                   ; 2370
                                    7E  D4 00317         CLRL    -(SP)
                       0004845A     8F  DD 00319         PUSHL   #296026
       00000000G  00                03  FB 0031F         CALLS   #3, LIB$SIGNAL
                                    52  DD 00326 28$:    PUSHL   SJH_N                                    ; 2371
       00000000G  EF                01  FB 00328         CALLS   #1, RELEASE_RECORD
                                    01  DD 0032F 29$:    PUSHL   #1                                       ; 2377
       00000000G  EF                01  FB 00331         CALLS   #1, REWRITE_RECORD
                  52                57  D0 00338         MOVL    R7, P                                    ; 2382
                                    0E  11 0033B         BRB     31$
                                    52  DD 0033D 30$:    PUSHL   P                                        ; 2383
       00000000G  EF                01  FB 0033F         CALLS   #1, DEALLOCATE_MEMORY
                  52       FE00     C2  9E 00346         MOVAB   -512(R2), P
             08   AE                52  D1 0034B 31$:    CMPL    P, BITMAP_RETADR
                                    EC  1E 0034F         BGEQU   30$
                                        04 00351         RET                                              ; 2384

; Routine Size:  850 bytes,    Routine Base:  CODE + 0669
```

```
;  1352     2385  1 ROUTINE COLD_START_NEW_FILE: NOVALUE=
;  1353     2386  1
;  1354     2387  1 !++
;  1355     2388  1 !
;  1356     2389  1 ! FUNCTIONAL DESCRIPTION:
;  1357     2390  1 !       This routine initializes a newly created queue file.
;  1358     2391  1 !
;  1359     2392  1 ! INPUT PARAMETERS:
;  1360     2393  1 !       NONE
;  1361     2394  1 !
;  1362     2395  1 ! IMPLICIT INPUTS:
;  1363     2396  1 !       NONE
;  1364     2397  1 !
;  1365     2398  1 ! OUTPUT PARAMETERS:
;  1366     2399  1 !       NONE
;  1367     2400  1 !
;  1368     2401  1 ! IMPLICIT OUTPUTS:
;  1369     2402  1 !       NONE
;  1370     2403  1 !
;  1371     2404  1 ! ROUTINE VALUE:
;  1372     2405  1 !       NONE
;  1373     2406  1 !
;  1374     2407  1 ! SIDE EFFECTS:
;  1375     2408  1 !       NONE
;  1376     2409  1 !
;  1377     2410  1 !--
;  1378     2411  1
;  1379     2412  2 BEGIN
;  1380     2413  2 LOCAL
;  1381     2414  2         BUFFER:         BBLOCK[SYM$S_SYM];      ! Record buffer
;  1382     2415  2
;  1383     2416  2
;  1384     2417  2 ! Link all records to the free list.
;  1385     2418  2 !
;  1386     2419  2 CH$FILL(0, SYM$S_SYM, BUFFER);
;  1387     2420  2 QUEUE_RAB[RAB$L_RBF] = BUFFER;
;  1388     2421  2 IF .FLAGS[FLAGS_V_QUEUE_SHARED]
;  1389     2422  2     THEN QUEUE_RAB[RAB$L_ROP] = RAB$M_UIF OR RAB$M_NLK
;  1390     2423  2     ELSE QUEUE_RAB[RAB$L_ROP] = RAB$M_UIF;
;  1391     2424  2 DECR REC_N FROM .QUEUE_FAB[FAB$L_ALQ] TO 4 DO
;  1392     2425  3     BEGIN
;  1393     2426  3     QUEUE_RAB[RAB$L_KBF] = REC_N;
;  1394     2427  3     DIAG_COUNT[1] = .DIAG_COUNT[1] + 1;
;  1395     2428  4     IF NOT $PUT(RAB=QUEUE_RAB)
;  1396     2429  3     THEN
;  1397     2430  3         SIGNAL_FILE_ERROR(
;  1398     2431  3             JBC$_WRITEERR + STS$K_SEVERE,
;  1399     2432  3             QUEUE_FAB, QUEUE_RAB);
;  1400     2433  3     BUFFER[SYM$L_LINK] = .REC_N;
;  1401     2434  2     END;
;  1402     2435  2
;  1403     2436  2
;  1404     2437  2 ! Initialize the queue header record.
;  1405     2438  2 !
;  1406     2439  2 CH$FILL(0, SYM$S_SYM, BUFFER);
;  1407     2440  2 BUFFER[SYM$B_TYPE] = SYM$K_SQH;
;  1408     2441  2 BUFFER[SQH$W_STRUCTURE_LEVEL] = SQH$K_STRUCTURE_LEVEL;
```

```
;  1409          2442  2 BUFFER[SQH$L_FORM_INDEX_LIST] = .SFX_KBF;
;  1410          2443  2 BUFFER[SQH$L_FREE_LIST] = 4;
;  1411          2444  2 BUFFER[SQH$L_HIGHEST_RECORD] = .QUEUE_FAB[FAB$L_ALQ];
;  1412          2445  2 BUFFER[SQH$L_NEXT_ENTRY_NUMBER] = 1;
;  1413          2446  2 BUFFER[SQH$L_HIGHEST_ENTRY_NUMBER] = SQH$S_ENTRY_BITMAP * 8;
;  1414          2447  2 QUEUE_RAB[RAB$L_KBF] = SQH_KBF;
;  1415          2448  2 DIAG_COUNT[1] = .DIAG_COUNT[1] + 1;
;  1416          2449  3 IF NOT $PUT(RAB=QUEUE_RAB)
;  1417          2450  2 THEN
;  1418          2451  2     SIGNAL_FILE_ERROR(
;  1419          2452  2         JBC$_WRITEERR + STS$K_SEVERE,
;  1420          2453  2         QUEUE_FAB, QUEUE_RAB);
;  1421          2454  2
;  1422          2455  2
;  1423          2456  2 ! Initialize the default form definition record.
;  1424          2457  2 !
;  1425          2458  2 CH$FILL(0, SYM$S_SYM, BUFFER);
;  1426          2459  2 BUFFER[SYM$B_TYPE] = SYM$K_SFM;
;  1427          2460  2 BUFFER[SFM$V_TRUNCATE] = TRUE;
;  1428          2461  2 BUFFER[SFM$B_LENGTH] = 66;
;  1429          2462  2 BUFFER[SFM$B_MARGIN_BOTTOM] = 6;
;  1430          2463  2 CH$MOVE(
;  1431          2464  2     %CHARCOUNT('DEFAULT') + 1,
;  1432          2465  2     UPLIT BYTE (%ASCIC 'DEFAULT'),
;  1433          2466  2     BUFFER[SFM$T_NAME]);
;  1434          2467  2 CH$MOVE(
;  1435          2468  2     %CHARCOUNT('System-defined default') + 1,
;  1436          2469  2     UPLIT BYTE (%ASCIC 'System-defined default'),
;  1437          2470  2     BUFFER[SFM$T_DESCRIPTION]);
;  1438          2471  2 CH$MOVE(SFM$S_NAME, BUFFER[SFM$T_NAME], BUFFER[SFM$T_STOCK]);
;  1439          2472  2 BUFFER[SFM$W_WIDTH] = 132;
;  1440          2473  2 QUEUE_RAB[RAB$L_KBF] = SFM_KBF;
;  1441          2474  2 DIAG_COUNT[1] = .DIAG_COUNT[1] + 1;
;  1442          2475  3 IF NOT $PUT(RAB=QUEUE_RAB)
;  1443          2476  2 THEN
;  1444          2477  2     SIGNAL_FILE_ERROR(
;  1445          2478  2         JBC$_WRITEERR + STS$K_SEVERE,
;  1446          2479  2         QUEUE_FAB, QUEUE_RAB);
;  1447          2480  2
;  1448          2481  2
;  1449          2482  2 ! Initialize the initial form index record.
;  1450          2483  2 !
;  1451          2484  2 CH$FILL(0, SYM$S_SYM, BUFFER);
;  1452          2485  2 BUFFER[SYM$B_TYPE] = SYM$K_SFX;
;  1453          2486  2 CH$MOVE(
;  1454          2487  2     %CHARCOUNT('DEFAULT') + 1,
;  1455          2488  2     UPLIT BYTE (%ASCIC 'DEFAULT'),
;  1456          2489  2     BBLOCK[BUFFER[SYM$T_DATA], SFX$T_NAME]);
;  1457          2490  2 BBLOCK[BUFFER[SYM$T_DATA], SFX$L_FORM_LINK] = .SFM_KBF;
;  1458          2491  2 QUEUE_RAB[RAB$L_KBF] = SFX_KBF;
;  1459          2492  2 DIAG_COUNT[1] = .DIAG_COUNT[1] + 1;
;  1460          2493  3 IF NOT $PUT(RAB=QUEUE_RAB)
;  1461          2494  2 THEN
;  1462          2495  2     SIGNAL_FILE_ERROR(
;  1463          2496  2         JBC$_WRITEERR + STS$K_SEVERE,
;  1464          2497  2         QUEUE_FAB, QUEUE_RAB);
;  1465          2498  2
```

```
;  1466          2499  2
;  1467          2500  2 ! Update the end of file pointer.
;  1468          2501  2 !
;  1469          2502  3 IF NOT $FLUSH(RAB=QUEUE_RAB)
;  1470          2503  2 THEN
;  1471          2504  2     SIGNAL_FILE_ERROR(
;  1472          2505  2         JBC$_WRITEERR + STS$K_SEVERE,
;  1473          2506  2         QUEUE_FAB, QUEUE_RAB);
;  1474          2507  1 END;


                                54  4C  55  41  46  45  44  07  009BB P.AAD:  .ASCII  <7>\DEFAULT\
64  65  6E  69  66  65  64       2D  6D  65  74  73  79  53  16  009C3 P.AAE:  .ASCII  <22>\System-defined default\
                                74  6C  75  61  66  65  64  20  009D2
                                54  4C  55  41  46  45  44  07  009DA P.AAF:  .ASCII  <7>\DEFAULT\


                                                    03FC 00000 COLD_START_NEW_FILE:
                                                               .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9                ; 2385
                                59      F620    CF  9E 00002       MOVAB   SFX_KBF, R9
                                58 00000000G    00  9E 00007       MOVAB   SYS$PUT, R8
                                57 00000000G    EF  9E 0000E       MOVAB   SIGNAL_FILE_ERROR, R7
                                56 00000000'    EF  9E 00015       MOVAB   QUEUE_RAB, R6
                                5E      FDFC    CE  9E 0001C       MOVAB   -516(SP), SP
    0200    8F          00          6E          00  2C 00021       MOVC5   #0, (SP), #0, #512, BUFFER                  ; 2419
                                            04  AE     00028
                                28  A6        04  AE  9E 0002A       MOVAB   BUFFER, QUEUE_RAB+40                    ; 2420
                            0A  FE9C    C6          03  E1 0002F       BBC     #3, FLAGS, 1$                       ; 2421
                                04  A6 00100010    8F  D0 00035       MOVL    #1048592, QUEUE_RAB+4               ; 2422
                                                    04  11 0003D       BRB     2$
                                04  A6            10  D0 0003F 1$:    MOVL    #16, QUEUE_RAB+4                    ; 2423
                                6E          C0  A6  D0 00043 2$:    MOVL    QUEUE_FAB+16, REC_N                 ; 2424
                                                    24  11 00047       BRB     5$
                                30  A6            6E  9E 00049 3$:    MOVAB   REC_N, QUEUE_RAB+48                 ; 2426
                                        FD00    C6  D6 0004D       INCL    DIAG_COUNT+4                        ; 2427
                                                    56  DD 00051       PUSHL   R6                                  ; 2428
                                    68              01  FB 00053       CALLS   #1, SYS$PUT
                                    0E              50  E8 00056       BLBS    R0, 4$
                                                    56  DD 00059       PUSHL   R6                                  ; 2430
                                        B0      A6  9F 0005B       PUSHAB  QUEUE_FAB
                                00410D4         8F  DD 0005E       PUSHL   #266452                             ; 2431
                                    67              03  FB 00064       CALLS   #3, SIGNAL_FILE_ERROR
                                04  AE            6E  D0 00067 4$:    MOVL    REC_N, BUFFER                       ; 2433
                                                    6E  D7 0006B       DECL    REC_N                               ; 2424
                                    04              6E  D1 0006D 5$:    CMPL    REC_N, #4
                                                    D7  18 00070       BGEQ    3$
0200    8F          00          6E          00  2C 00072       MOVC5   #0, (SP), #0, #512, BUFFER                  ; 2439
                                            04  AE     00079
                                08  AE            04  90 0007B       MOVB    #4, BUFFER+4                        ; 2440
                                10  AE      0401  8F  B0 0007F       MOVW    #1025, BUFFER+12                    ; 2441
                                38  AE            69  D0 00085       MOVL    SFX_KBF, BUFFER+52                  ; 2442
                                3C  AE            04  D0 00089       MOVL    #4, BUFFER+56                       ; 2443
                                44  AE        C0  A6  D0 0008D       MOVL    QUEUE_FAB+16, BUFFER+64             ; 2444
                                4C  AE            01  D0 00092       MOVL    #1, BUFFER+72                       ; 2445
                                40  AE      0800  8F  3C 00096       MOVZWL  #2048, BUFFER+60                    ; 2446
```

```
                          30   A6        F8   A9  9E 0009C          MOVAB   SQH_KBF, QUEUE_RAB+48          ; 2447
                                     FD00   C6  D6 000A1          INCL    DIAG_COUNT+4                   ; 2448
                                            56  DD 000A5          PUSHL   R6                             ; 2449
                                        68      01  FB 000A7          CALLS   #1, SYS$PUT                    ;
                                        0E      50  E8 000AA          BLBS    R0, 6$                         ;
                                            56  DD 000AD          PUSHL   R6                             ; 2451
                                                BC  A6  9F 000AF          PUSHAB  QUEUE_FAB                      ;
                                    00041DD4    8F  DD 000B2          PUSHL   #266452                        ; 2452
                                        67      03  FB 000B8          CALLS   #3, SIGNAL_FILE_ERROR          ;
    0200  8F      00                    6E      00  2C 000BB 6$:      MOVC5   #0, (SP), #0, #512, BUFFER     ; 2458
                                            04  AE      000C2                                              ;
                              08        AE      02  90 000C4          MOVB    #2, BUFFER+4                   ; 2459
                              10        AE      02  8E 000C8          BISB2   #2, BUFFER+12                  ; 2460
                            FF5A        CD  0642    8F  B0 000CC          MOVW    #1602, BUFFER+346              ; 2461
                  FF10      CD  09B3    C9      08  28 000D3          MOVC3   #8, P.AAD, BUFFER+272          ; 2466
                    14      AE  09B8    C9      17  28 000DB          MOVC3   #23, P.AAE, BUFFER+16          ; 2470
                  FF34      CD  FF10    CD      20  28 000E2          MOVC3   #32, BUFFER+272, BUFFER+308    ; 2471
                            FF58        CD  84  8F  9B 000EA          MOVZBW  #132, BUFFER+344               ; 2472
                              30        A6      FC  A9  9E 000F0          MOVAB   SFM_KBF, QUEUE_RAB+48          ; 2473
                                     FD00   C6  D6 000F5          INCL    DIAG_COUNT+4                   ; 2474
                                            56  DD 000F9          PUSHL   R6                             ; 2475
                                        68      01  FB 000FB          CALLS   #1, SYS$PUT                    ;
                                        0E      50  E8 000FE          BLBS    R0, 7$                         ;
                                            56  DD 00101          PUSHL   R6                             ; 2477
                                                B0  A6  9F 00103          PUSHAB  QUEUE_FAB                      ;
                                    000410D4    8F  DD 00106          PUSHL   #266452                        ; 2478
                                        67      03  FB 0010C          CALLS   #3, SIGNAL_FILE_ERROR          ;
    0200  8F      00                    6E      00  2C 0010F 7$:      MOVC5   #0, (SP), #0, #512, BUFFER     ; 2484
                                            04  AE      00116                                              ;
                              08        AE      03  90 00118          MOVB    #3, BUFFER+4                   ; 2485
                      10      AE  09D2    C9      08  28 0011C          MOVC3   #8, P.AAF, BUFFER+12           ; 2489
                              34        AE      FC  A9  D0 00123          MOVL    SFM_KBF, BUFFER+48             ; 2490
                              30        A6      69  9E 00128          MOVAB   SFX_KBF, QUEUE_RAB+48          ; 2491
                                     FD00   C6  D6 0012C          INCL    DIAG_COUNT+4                   ; 2492
                                            56  DD 00130          PUSHL   R6                             ; 2493
                                        68      01  FB 00132          CALLS   #1, SYS$PUT                    ;
                                        0E      50  E8 00135          BLBS    R0, 8$                         ;
                                            56  DD 00138          PUSHL   R6                             ; 2495
                                                B0  A6  9F 0013A          PUSHAB  QUEUE_FAB                      ;
                                    000410D4    8F  DD 0013D          PUSHL   #266452                        ; 2496
                                        67      03  FB 00143          CALLS   #3, SIGNAL_FILE_ERROR          ;
                                            56  DD 00146 8$:      PUSHL   R6                             ; 2502
                        00000000G   00          01  FB 00148          CALLS   #1, SYS$FLUSH                  ;
                                        0E      50  E8 0014F          BLBS    R0, 9$                         ;
                                            56  DD 00152          PUSHL   R6                             ; 2504
                                                B0  A6  9F 00154          PUSHAB  QUEUE_FAB                      ;
                                    000410D4    8F  DD 00157          PUSHL   #266452                        ; 2505
                                        67      03  FB 0015D          CALLS   #3, SIGNAL_FILE_ERROR          ;
                                                    04 00160 9$:      RET                                    ; 2507

; Routine Size:  353 bytes,    Routine Base:  CODE + 09E2
```

```
;  1476     2508  1 GLOBAL ROUTINE WARM_START_EXISTING_FILE(SYSID): NOVALUE=
;  1477     2509  1
;  1478     2510  1 !++
;  1479     2511  1 !
;  1480     2512  1 ! FUNCTIONAL DESCRIPTION:
;  1481     2513  1 !       This routine reinitializes an existing queue file if this job controller
;  1482     2514  1 !       is not the first accessor.
;  1483     2515  1 !
;  1484     2516  1 ! INPUT PARAMETERS:
;  1485     2517  1 !       SYSID           - Pointer to system ID.
;  1486     2518  1 !
;  1487     2519  1 ! IMPLICIT INPUTS:
;  1488     2520  1 !       NONE
;  1489     2521  1 !
;  1490     2522  1 ! OUTPUT PARAMETERS:
;  1491     2523  1 !       NONE
;  1492     2524  1 !
;  1493     2525  1 ! IMPLICIT OUTPUTS:
;  1494     2526  1 !       NONE
;  1495     2527  1 !
;  1496     2528  1 ! ROUTINE VALUE:
;  1497     2529  1 !       NONE
;  1498     2530  1 !
;  1499     2531  1 ! SIDE EFFECTS:
;  1500     2532  1 !       NONE
;  1501     2533  1 !
;  1502     2534  1 !--
;  1503     2535  1
;  1504     2536  2 BEGIN
;  1505     2537  2 LOCAL
;  1506     2538  2         SQH:            REF BBLOCK,     ! Pointer to SQH
;  1507     2539  2         SQX:            REF BBLOCK,     ! Pointer to SQX
;  1508     2540  2         SQX_N,                          ! Record number of SQX
;  1509     2541  2         SQX_NS,                         ! Record number of successor of SQX
;  1510     2542  2         SQE:            REF BBLOCK,     ! Pointer to SQX entry
;  1511     2543  2         SMQ_N,                          ! Record number of SMQ
;  1512     2544  2         SMQ:            REF BBLOCK,     ! Pointer to SMQ
;  1513     2545  2         SJH_NP,                         ! Record number of predecessor of SJH
;  1514     2546  2         SJH_P:          REF BBLOCK,     ! Pointer to predecessor of SJH
;  1515     2547  2         SJH_N,                          ! Record number of SJH
;  1516     2548  2         SJH:            REF BBLOCK,     ! Pointer to SJH
;  1517     2549  2         SJH_NS;                         ! Record number of successor of SJH
;  1518     2550  2
;  1519     2551  2
;  1520     2552  2 ! Read the queue header.
;  1521     2553  2 !
;  1522     2554  2 SQH = READ_RECORD(SQH$K_RECNO);
;  1523     2555  2
;  1524     2556  2
;  1525     2557  2 ! Reinitialize all executor queues that were assigned to the failing or
;  1526     2558  2 ! restarting node.
;  1527     2559  2 !
;  1528     2560  2 SQX_N = .SQH[SQH$L_QUEUE_INDEX_LIST];
;  1529     2561  2 WHILE .SQX_N NEQ 0 DO
;  1530     2562  3     BEGIN
;  1531     2563  3
;  1532     2564  3     ! Read the queue index record.
```

```
;  1533          2565  3      !
;  1534          2566  3      SQX = READ_RECORD(.SQX_N);
;  1535          2567  3
;  1536          2568  3
;  1537          2569  3      ! Scan the queue index record.
;  1538          2570  3      !
;  1539          2571  3      SQE = SQX[SYM$T_DATA];
;  1540          2572  3      INCR SQE_N FROM 0 TO SQX$K_ENTRIES-1 DO
;  1541          2573  4          BEGIN
;  1542          2574  4          IF CH$RCHAR(SQE[SQX$T_NAME]) EQL 0
;  1543          2575  4          THEN
;  1544          2576  4              EXITLOOP
;  1545          2577  4          ELSE
;  1546          2578  5              BEGIN
;  1547          2579  5              IF .SQE[SQX$V_EXECUTOR]
;  1548          2580  5              THEN
;  1549          2581  6                  BEGIN
;  1550          2582  6                  SMQ = READ_RECORD(SMQ_N = .SQE[SQX$L_QUEUE_LINK]);
;  1551          2583  7                  IF SYSID_EQL(.SYSID, SMQ[SMQ$T_SYSID])
;  1552          2584  6                  THEN
;  1553          2585  7                      BEGIN
;  1554          2586  7
;  1555          2587  7                      ! Reinitialize the queue header as appropriate.
;  1556          2588  7                      !
;  1557          2589  7                      SMQ[SMQ$L_STATUS] = 0;
;  1558          2590  7                      SMQ[SMQ$V_STOPPED] = TRUE;
;  1559          2591  7                      CLEAR_TIME(SMQ[SMQ$Q_ACM_BEGTIM]);
;  1560          2592  7                      SMQ[SMQ$L_ACM_GETCNT] = 0;
;  1561          2593  7                      SMQ[SMQ$L_ACM_PAGECNT] = 0;
;  1562          2594  7                      SMQ[SMQ$L_ACM_QIOCNT] = 0;
;  1563          2595  7                      SMQ[SMQ$L_ACM_SYMCPUTIM] = 0;
;  1564          2596  7                      SMQ[SMQ$L_STREAM_SCT] = 0;
;  1565          2597  7                      SMQ[SMQ$B_CURRENT_JOB_COUNT] = 0;
;  1566          2598  7                      SMQ[SMQ$B_STREAM_INDEX] = 0;
;  1567          2599  7                      REWRITE_RECORD(.SMQ_N);
;  1568          2600  7                      END
;  1569          2601  6                  ELSE
;  1570          2602  6                      RELEASE_RECORD(.SMQ_N);
;  1571          2603  5                  END;
;  1572          2604  4              END;
;  1573          2605  4
;  1574          2606  4
;  1575          2607  4          SQE = .SQE + SQX$S_SQX;
;  1576          2608  3          END;
;  1577          2609  3
;  1578          2610  3
;  1579          2611  3      ! Advance to the next index block.
;  1580          2612  3      !
;  1581          2613  3      SQX_NS = .SQX[SYM$L_LINK];
;  1582          2614  3      RELEASE_RECORD(.SQX_N);
;  1583          2615  3      SQX_N = .SQX_NS;
;  1584          2616  2      END;
;  1585          2617  2
;  1586          2618  2
;  1587          2619  2  ! Requeue or delete jobs that were executing on the failed or restarting queues.
;  1588          2620  2  !
;  1589          2621  2  SQX_N = .SQH[SQH$L_QUEUE_INDEX_LIST];
```

```
;  1590          2622  2 WHILE .SQX_N NEQ 0 DO
;  1591          2623  3     BEGIN
;  1592          2624  3
;  1593          2625  3     ! Read the queue index record.
;  1594          2626  3     !
;  1595          2627  3     SQX = READ_RECORD(.SQX_N);
;  1596          2628  3
;  1597          2629  3
;  1598          2630  3     ! Scan the queue index record.
;  1599          2631  3     !
;  1600          2632  3     SQE = SQX[SYM$T_DATA];
;  1601          2633  3     INCR SQE_N FROM 0 TO SQX$K_ENTRIES-1 DO
;  1602          2634  4         BEGIN
;  1603          2635  4         IF CH$RCHAR(SQE[SQX$T_NAME]) EQL 0
;  1604          2636  4         THEN
;  1605          2637  4             EXITLOOP
;  1606          2639  4         ELSE
;  1607          2640  5             BEGIN
;  1608          2641  5             IF .SQE[SQX$V_EXECUTOR]
;  1609          2642  5             THEN
;  1610          2643  6                 BEGIN
;  1611          2644  6                 SMQ = READ_RECORD(SMQ_N = .SQE[SQX$L_QUEUE_LINK]);
;  1612          2645  7                 IF SYSID_EQL(.SYSID, SMQ[SMQ$T_SYSID])
;  1613          2646  6                 THEN
;  1614          2647  7                     BEGIN
;  1615          2648  7
;  1616          2649  7                     ! Requeue or delete each job, as appropriate.
;  1617          2650  7                     !
;  1618          2651  8                     WHILE .SMQ[SMQ$L_CURRENT_LIST] NEQ 0 DO
;  1619          2652  8                         BEGIN
;  1620          2653  8                         LOCAL
;  1621          2654  8                             SJH_N,                  ! Record number of SJH
;  1622          2655  8                             SJH:        REF BBLOCK; ! Pointer to SJH
;  1623          2656  8
;  1624          2657  8                         SJH = READ_RECORD(SJH_N = .SMQ[SMQ$L_CURRENT_LIST]);
;  1625          2658  8
;  1626          2659  8                         ! Check for a deallocated SJH on the current list.
;  1627          2660  8                         !
;  1628          2661  8                         IF .SJH[SJH$L_QUEUE_LINK] EQL 0
;  1629          2662  8                         THEN
;  1630          2663  9                             BEGIN
;  1631          2664  9                             SMQ[SMQ$L_CURRENT_LIST] = 0;
;  1632          2665  9                             EXITLOOP;
;  1633          2666  8                             END;
;  1634          2667  8
;  1635          2668  8                         UPDATE_GETQUI_DATA(.SJH_N, .SJH);
;  1636          2669  8                         SMQ[SMQ$L_CURRENT_LIST] = .SJH[SYM$L_LINK];
;  1637          2670  8
;  1638          2671  8                         ! Flush the SMQ so as not to corrupt current list.
;  1639          2672  8                         !
;  1640          2673  8                         FLUSH_RECORD(.SMQ_N);
;  1641          2674  8                         SJH[SJH$V_SYSTEM_FAILURE] = TRUE;
;  1642          2675  8                         COMPLETE_JOB(
;  1643          2676  8                             .SJH_N, .SJH,
;  1644          2677  8                             .SMQ,
;  1645          2678  8                             0,
;  1646
```

```
;  1647     2679  8                          JBC$_SYSFAIL OR STS$K_ERROR);
;  1648     2680  7                      END;
;  1649     2681  7                  SMQ[SMQ$L_CURRENT_LIST_END] = 0;
;  1650     2682  7                  REWRITE_RECORD(.SMQ_N);
;  1651     2683  7                  END
;  1652     2684  6              ELSE
;  1653     2685  6                  RELEASE_RECORD(.SMQ_N);
;  1654     2686  5              END;
;  1655     2687  4          END;
;  1656     2688  4
;  1657     2689  4
;  1658     2690  4
;  1659     2691  4      SQE = .SQE + SQX$S_SQX;
;  1660     2692  3      END;
;  1661     2693  3
;  1662     2694  3
;  1663     2695  3      ! Advance to the next index block.
;  1664     2696  3      !
;  1665     2697  3      SQX_NS = .SQX[SYM$L_LINK];
;  1666     2698  3      RELEASE_RECORD(.SQX_N);
;  1667     2699  3      SQX_N = .SQX_NS;
;  1668     2700  2      END;
;  1669     2701  2
;  1670     2702  2
;  1671     2703  2 ! Purge references to this system from the incomplete service list.
;  1672     2704  2 !
;  1673     2705  2 SCAN_INCOMPLETE_SERVICES(ISRV_K_PURGE_SYSID, .SYSID);
;  1674     2706  2
;  1675     2707  2
;  1676     2708  2 ! Purge open jobs for this system from the open job queue.
;  1677     2709  2 !
;  1678     2710  2 SJH_NP = 0;
;  1679     2711  2 SJH_N = .SQH[SQH$L_OPEN_LIST];
;  1680     2712  2 WHILE .SJH_N NEQ 0 DO
;  1681     2713  3     BEGIN
;  1682     2714  3     SJH = READ_RECORD(.SJH_N);
;  1683     2715  3     SJH_NS = .SJH[SYM$L_LINK];
;  1684     2716  3
;  1685     2717  3
;  1686     2718  4     IF SYSID_EQL(.SYSID, SJH[SJH$T_SYSID])
;  1687     2719  3     THEN
;  1688     2720  4         BEGIN
;  1689     2721  4         IF .SJH_NP EQL 0
;  1690     2722  4         THEN
;  1691     2723  5             BEGIN
;  1692     2724  5             SQH[SQH$L_OPEN_LIST] = .SJH[SYM$L_LINK];
;  1693     2725  5             IF .SJH[SYM$L_LINK] EQL 0 THEN SQH[SQH$L_OPEN_LIST_END] = 0;
;  1694     2726  5             END
;  1695     2727  4         ELSE
;  1696     2728  5             BEGIN
;  1697     2729  5             SJH_P[SYM$L_LINK] = .SJH[SYM$L_LINK];
;  1698     2730  5             IF .SJH[SYM$L_LINK] EQL 0 THEN SQH[SQH$L_OPEN_LIST_END] = .SJH_NP;
;  1699     2731  4             END;
;  1700     2732  4         SMQ = READ_RECORD(SMQ_N = .SJH[SJH$L_QUEUE_LINK]);
;  1701     2733  4         SMQ[SMQ$W_OPEN_JOB_COUNT] = .SMQ[SMQ$W_OPEN_JOB_COUNT] - 1;
;  1702     2734  4         REWRITE_RECORD(.SMQ_N);
;  1703     2735  4         DELETE_SJH_RECORD(.SJH_N, .SJH);
```

```
;  1704          2736  4          END
;  1705          2737  3      ELSE
;  1706          2738  4          BEGIN
;  1707          2739  4          IF .SJH_NP NEQ 0 THEN REWRITE_RECORD(.SJH_NP);
;  1708          2740  4          SJH_NP = .SJH_N;
;  1709          2741  4          SJH_P = .SJH;
;  1710          2742  3          END;
;  1711          2743  3      SJH_N = .SJH_NS;
;  1712          2744  2      END;
;  1713          2745  2  IF .SJH_NP NEQ 0 THEN REWRITE_RECORD(.SJH_NP);
;  1714          2746  2
;  1715          2747  2
;  1716          2748  2  ! If the timer list is not empty, set a timer on the first job.
;  1717          2749  2  !
;  1718          2750  2  IF .SQH[SQH$L_TIMER_LIST] NEQ 0
;  1719          2751  2  THEN
;  1720          2752  3      BEGIN
;  1721          2753  3      LOCAL
;  1722          2754  3          SJH_N,                          ! Record number of SJH
;  1723          2755  3          SJH:            REF BBLOCK,     ! Pointer to SJH
;  1724          2756  3          STATUS;                         ! Status return
;  1725          2757  3
;  1726          2758  3      SJH = READ_RECORD(SJH_N = .SQH[SQH$L_TIMER_LIST]);
;  1727        P 2759  3      STATUS = $SETIMR(
;  1728        P 2760  3          DAYTIM=SJH[SJH$Q_AFTER_TIME],
;  1729        P 2761  3          ASTADR=AFTER_AST,
;  1730          2762  3          REQIDT=JBC$K_AFTER_IDT);
;  1731          2763  3      IF NOT .STATUS
;  1732          2764  3      THEN
;  1733          2765  3          SIGNAL(JBC$_SETIMR OR STS$K_ERROR, 0, .STATUS);
;  1734          2766  3      RELEASE_RECORD(.SJH_N);
;  1735          2767  2      END;
;  1736          2768  2
;  1737          2769  2
;  1738          2770  2  ! Rewrite the queue header.
;  1739          2771  2  !
;  1740          2772  2  REWRITE_RECORD(SQH$K_RECNO);
;  1741          2773  1  END;
;  INFO#250                L1:2729
;  Referenced LOCAL symbol SJH_P is probably not initialized
```

```
                                    0FFC 00000          .ENTRY  WARM_START_EXISTING_FILE, Save R2,R3,R4,R5,-  ; 2508
                                                                R6,R7,R8,R9,R10,R11                           ;
                          5E              04  C2 00002          SUBL2   #4, SP                                ;
                                          01  DD 00005          PUSHL   #1                                    ; 2554
                00000000G EF              01  FB 00007          CALLS   #1, READ_RECORD                       ;
                          55              50  D0 0000E          MOVL    R0, SQH                               ;
                          58        64    A5  D0 00011          MOVL    100(SQH), SQX_N                       ; 2560
                                          03  12 00015  1$:      BNEQ    2$                                    ; 2561
                                      0088  31 00017          BRW     7$                                    ;
                                          58  DD 0001A  2$:      PUSHL   SQX_N                                 ; 2566
                00000000G EF              01  FB 0001C          CALLS   #1, READ_RECORD                       ;
                          6E              50  D0 00023          MOVL    R0, SQX                               ;
```

```
                54              6E              0C  C1 00026          ADDL3   #12, SQX, SQE                   ; 2571
                                                53  D4 0002A          CLRL    SQE_N                           ; 2572
                                                64  95 0002C 3$:      TSTB    (SQE)                           ; 2574
                                                5F  13 0002E          BEQL    6$                              ;
                53      20      A4              01  E1 00030          BBC     #1, 32(SQE), 5$                 ; 2579
                                59      24      A4  D0 00035          MOVL    36(SQE), SMQ_N                  ; 2582
                                                59  DD 00039          PUSHL   SMQ_N                           ;
                    00000000G   EF              01  FB 0003B          CALLS   #1, READ_RECORD                 ;
                                52              50  D0 00042          MOVL    R0, SMQ                         ;
                                50      04      AC  D0 00045          MOVL    SYSID, R0                       ; 2583
                        0106    C2              60  D1 00049          CMPL    (R0), 262(SMQ)                  ;
                                                2F  12 0004E          BNEQ    4$                              ;
                        010A    C2      04      A0  B1 00050          CMPW    4(R0), 266(SMQ)                 ;
                                                27  12 00056          BNEQ    4$                              ;
                                        10      A2  D4 00058          CLRL    16(SMQ)                         ; 2589
                                11      A2          02  88 0005B      BISB2   #2, 17(SMQ)                     ; 2590
                                        14      A2  7C 0005F          CLRQ    20(SMQ)                         ; 2591
                                        1C      A2  7C 00062          CLRQ    28(SMQ)                         ; 2592
                                        24      A2  7C 00065          CLRQ    36(SMQ)                         ; 2594
                                    00FC        C2  D4 00068          CLRL    252(SMQ)                        ; 2596
                                    0115        C2  94 0006C          CLRB    277(SMQ)                        ; 2597
                                    0117        C2  94 00070          CLRB    279(SMQ)                        ; 2598
                                                59  DD 00074          PUSHL   SMQ_N                           ; 2599
                    00000000G   EF              01  FB 00076          CALLS   #1, REWRITE_RECORD              ;
                                                09  11 0007D          BRB     5$                              ; 2583
                                                59  DD 0007F 4$:      PUSHL   SMQ_N                           ; 2602
                    00000000G   EF              01  FB 00081          CALLS   #1, RELEASE_RECORD              ;
                                54              28  C0 00088 5$:      ADDL2   #40, SQE                        ; 2607
                9D              53              0B  F3 0008B          AOBLEQ  #11, SQE_N, 3$                  ; 2572
                                5B      00      BE  D0 0008F 6$:      MOVL    @SQX, SQX_NS                    ; 2613
                                                58  DD 00093          PUSHL   SQX_N                           ; 2614
                    00000000G   EF              01  FB 00095          CALLS   #1, RELEASE_RECORD              ;
                                58              5B  D0 0009C          MOVL    SQX_NS, SQX_N                   ; 2615
                                            FF73    31 0009F          BRW     1$                              ; 2561
                                58      64      A5  D0 000A2 7$:      MOVL    100(SQH), SQX_N                 ; 2621
                                                03  12 000A6 8$:      BNEQ    9$                              ; 2622
                                            00CB    31 000A8          BRW     19$                             ;
                                                58  DD 000AB 9$:      PUSHL   SQX_N                           ; 2627
                    00000000G   EF              01  FB 000AD          CALLS   #1, READ_RECORD                 ;
                                6E              50  D0 000B4          MOVL    R0, SQX                         ;
                54              6E              0C  C1 000B7          ADDL3   #12, SQX, SQE                   ; 2632
                                                5A  D4 000BB          CLRL    SQE_N                           ; 2633
                                                64  95 000BD 10$:     TSTB    (SQE)                           ; 2635
                                                03  12 000BF          BNEQ    11$                             ;
                                            009F    31 000C1          BRW     18$                             ;
                03      20      A4              01  E0 000C4 11$:     BBS     #1, 32(SQE), 12$                ; 2640
                                            008E    31 000C9          BRW     17$                             ;
                                59      24      A4  D0 000CC 12$:     MOVL    36(SQE), SMQ_N                  ; 2643
                                                59  DD 000D0          PUSHL   SMQ_N                           ;
                    00000000G   EF              01  FB 000D2          CALLS   #1, READ_RECORD                 ;
                                52              50  D0 000D9          MOVL    R0, SMQ                         ;
                                50      04      AC  D0 000DC          MOVL    SYSID, R0                       ; 2644
                        0106    C2              60  D1 000E0          CMPL    (R0), 262(SMQ)                  ;
                                                6A  12 000E5          BNEQ    16$                             ;
                        010A    C2      04      A0  B1 000E7          CMPW    4(R0), 266(SMQ)                 ;
                                                62  12 000ED          BNEQ    16$                             ;
                                56      48      A2  9E 000EF          MOVAB   72(SMQ), R6                     ; 2650
```

```
                                                66  D5 000F3 13$:   TSTL    (R6)                                    ;
                                                4C  13 000F5        BEQL    15$                                     ;
                                             57     66  D0 000F7        MOVL    (R6), SJH_N                             ; 2657
                                                57  DD 000FA        PUSHL   SJH_N                                   ;
                                00000000G    EF     01  FB 000FC        CALLS   #1, READ_RECORD                         ;
                                             53     50  D0 00103        MOVL    R0, SJH                                 ;
                                      0134   C3     D5  00106        TSTL    308(SJH)                                ; 2661
                                                04  12 0C10A        BNEQ    14$                                     ;
                                                66  D4 0010C        CLRL    (R6)                                    ; 2664
                                                33  11 0010E        BRB     15$                                     ; 2663
                                                53  DD 00110 14$:   PUSHL   SJH                                     ; 2668
                                                57  DD 00112        PUSHL   SJH_N                                   ;
                                00000000G    EF     02  FB 00114        CALLS   #2, UPDATE_GETQUI_DATA                  ;
                                             66     63  D0 0011B        MOVL    (SJH), (R6)                             ; 2669
                                                59  DD 0011E        PUSHL   SMQ_N                                   ; 2673
                                00000000G    EF     01  FB 00120        CALLS   #1, FLUSH_RECORD                        ;
                                          11 A3        8F  88 00127        BISB2   #64, 17(SJH)                            ; 2674
                                      000480F2   8F     DD  0012C        PUSHL   #295154                                 ; 2679
                                                7E  D4 00132        CLRL    -(SP)                                   ; 2675
                                                52  DD 00134        PUSHL   SMQ                                     ; 2677
                                                53  DD 00136        PUSHL   SJH                                     ; 2676
                                                57  DD 00138        PUSHL   SJH_N                                   ;
                                00000000G    EF     05  FB 0013A        CALLS   #5, COMPLETE_JOB                        ;
                                                B0  11 00141        BRB     13$                                     ; 2650
                                      4C     A2  D4 00143 15$:   CLRL    76(SMQ)                                 ; 2681
                                                59  DD 00146        PUSHL   SMQ_N                                   ; 2682
                                00000000G    EF     01  FB 00148        CALLS   #1, REWRITE_RECORD                      ;
                                                09  11 0014F        BRB     17$                                     ; 2644
                                                59  DD 00151 16$:   PUSHL   SMQ_N                                   ; 2685
                                00000000G    EF     01  FB 00153        CALLS   #1, RELEASE_RECORD                      ;
                                             54     28  C0 0015A 17$:   ADDL2   #40, SQE                                ; 2691
FF5A                         5A              01     0B  F1 0015D        ACBL    #11, #1, SQE_N, 10$                     ; 2633
                                             5B  00 BE  D0 00163 18$:   MOVL    @SQX, SQX_NS                            ; 2697
                                                58  DD 00167        PUSHL   SQX_N                                   ; 2698
                                00000000G    EF     01  FB 00169        CALLS   #1, RELEASE_RECORD                      ;
                                             58     5B  D0 00170        MOVL    SQX_NS, SQX_N                           ; 2699
                                                FF30   31  00173        BRW     8$                                      ; 2622
                                             54  04 AC  D0 00176 19$:   MOVL    SYSID, R4                               ; 2705
                                                54  DD 0017A        PUSHL   R4                                      ;
                                                03  DD 0017C        PUSHL   #3                                      ;
                                00000000G    EF     02  FB 0017E        CALLS   #2, SCAN_INCOMPLETE_SERVICES            ;
                                                53  D4 00185        CLRL    SJH_NP                                  ; 2710
                                             56  4C A5  D0 00187        MOVL    76(SQH), SJH_N                          ; 2711
                                                79  13 0018B 20$:   BEQL    26$                                     ; 2712
                                                56  DD 0018D        PUSHL   SJH_N                                   ; 2714
                                00000000G    EF     01  FB 0018F        CALLS   #1, READ_RECORD                         ;
                                             58     50  D0 00196        MOVL    R0, SJH                                 ;
                                             57     68  D0 00199        MOVL    (SJH), SJH_NS                           ; 2715
                                   016C      C8     64  D1 0019C        CMPL    (R4), 364(SJH)                          ; 2718
                                                4B  12 001A1        BNEQ    23$                                     ;
                                   0170      C8  04 A4  B1 001A3        CMPW    4(R4), 368(SJH)                         ;
                                                43  12 001A9        BNEQ    23$                                     ;
                                                53  D5 001AB        TSTL    SJH_NP                                  ; 2721
                                                0B  12 001AD        BNEQ    21$                                     ;
                                      4C     A5     68  D0 001AF        MOVL    (SJH), 76(SQH)                          ; 2724
                                                0E  12 001B3        BNEQ    22$                                     ; 2725
                                             50  A5 04 D4 001B5        CLRL    80(SQH)                                 ;
```

```
                                   09  11 001B8           BRB     22$                              ; 2721
                    6A             68  D0 001BA  21$:     MOVL    (SJH), (SJH_P)                   ; 2729
                                   04  12 001BD           BNEQ    22$                              ; 2730
            50      A5             53  D0 001BF           MOVL    SJH_NP, 80(SQH)                  ;
                    59     0134    C8  D0 001C3  22$:     MOVL    308(SJH), SMQ_N                  ; 2732
                                   59  DD 001C8           PUSHL   SMQ_N                            ;
     00000000G      EF             01  FB 001CA           CALLS   #1, READ_RECORD                  ;
                    52             50  D0 001D1           MOVL    R0, SMQ                          ;
                           0100    C2  B7 001D4           DECW    256(SMQ)                         ; 2733
                                   59  DD 001D8           PUSHL   SMQ_N                            ; 2734
     00000000G      EF             01  FB 001DA           CALLS   #1, REWRITE_RECORD               ;
                           0140    8F  BB 001E1           PUSHR   #^M<R6,R8>                       ; 2735
     00000000G      EF             02  FB 001E5           CALLS   #2, DELETE_SJH_RECORD            ;
                                   13  11 001EC           BRB     25$                              ; 2718
                                   53  D5 001EE  23$:     TSTL    SJH_NP                           ; 2739
                                   09  13 001F0           BEQL    24$                              ;
                                   53  DD 001F2           PUSHL   SJH_NP                           ;
     00000000G      EF             01  FB 001F4           CALLS   #1, REWRITE_RECORD               ;
                    53             56  D0 001FB  24$:     MOVL    SJH_N, SJH_NP                    ; 2740
                    5A             58  D0 001FE           MOVL    SJH, SJH_P                       ; 2741
                    56             57  D0 00201  25$:     MOVL    SJH_NS, SJH_N                    ; 2743
                                   85  11 00204           BRB     20$                              ; 2712
                                   53  D5 00206  26$:     TSTL    SJH_NP                           ; 2745
                                   09  13 00208           BEQL    27$                              ;
                                   53  DD 0020A           PUSHL   SJH_NP                           ;
     00000000G      EF             01  FB 0020C           CALLS   #1, REWRITE_RECORD               ;
                             68    A5  D5 00213  27$:     TSTL    104(SQH)                         ; 2750
                                   3F  13 00216           BEQL    29$                              ;
                    52       68    A5  D0 00218           MOVL    104(SQH), SJH_N                  ; 2758
                                   52  DD 0021C           PUSHL   SJH_N                            ;
     00000000G      EF             01  FB 0021E           CALLS   #1, READ_RECORD                  ;
                                   01  DD 00225           PUSHL   #1                               ; 2762
                      00000000G    EF  9F 00227           PUSHAB  AFTER_AST                        ;
                           0098    C0  9F 0022D           PUSHAB  152(SJH)                         ;
                                   7E  D4 00231           CLRL    -(SP)                            ;
     00000000G      00             04  FB 00233           CALLS   #4, SYS$SETIMR                   ;
                    11             50  E8 0023A           BLBS    STATUS, 28$                      ; 2763
                                   50  DD 0023D           PUSHL   STATUS                           ; 2765
                                   7E  D4 0023F           CLRL    -(SP)                            ;
                      0004845A     8F  DD 00241           PUSHL   #296026                          ;
     00000000G      00             03  FB 00247           CALLS   #3, LIB$SIGNAL                   ;
                                   52  DD 0024E  28$:     PUSHL   SJH_N                            ; 2766
     00000000G      EF             01  FB 00250           CALLS   #1, RELEASE_RECORD               ;
                                   01  DD 00257  29$:     PUSHL   #1                               ; 2772
     00000000G      EF             01  FB 00259           CALLS   #1, REWRITE_RECORD               ;
                                       04 00260           RET                                      ; 2773

; Routine Size:  609 bytes,    Routine Base:  CODE + 0B43
```

```
;  1743          2774  1 ROUTINE INITIALIZE_QUEUE_FILE_HANDLER(SIG,MCH)=
;  1744          2775  1
;  1745          2776  1 !++
;  1746          2777  1 !
;  1747          2778  1 ! FUNCTIONAL DESCRIPTION:
;  1748          2779  1 !       This routine is a condition handler for INITIALIZE_QUEUE_FILE.  It
;  1749          2780  1 !       unwinds the stack and returns status if an error occurs.
;  1750          2781  1 !
;  1751          2782  1 ! INPUT PARAMETERS:
;  1752          2783  1 !       Standard VMS condition handler parameters.
;  1753          2784  1 !
;  1754          2785  1 ! IMPLICIT INPUTS:
;  1755          2786  1 !       NONE
;  1756          2787  1 !
;  1757          2788  1 ! OUTPUT PARAMETERS:
;  1758          2789  1 !       NONE
;  1759          2790  1 !
;  1760          2791  1 ! IMPLICIT OUTPUTS:
;  1761          2792  1 !       NONE
;  1762          2793  1 !
;  1763          2794  1 ! ROUTINE VALUE:
;  1764          2795  1 !       If the signal is a success status, SS$_RESIGNAL; otherwise an unwind
;  1765          2796  1 !       to caller of establisher with the failure status.
;  1766          2797  1 !
;  1767          2798  1 ! SIDE EFFECTS:
;  1768          2799  1 !       NONE
;  1769          2800  1 !
;  1770          2801  1 !--
;  1771          2802  1
;  1772          2803  2 BEGIN
;  1773          2804  2 MAP
;  1774          2805  2         SIG:            REF BBLOCK,     ! Pointer to signal vector
;  1775          2806  2         MCH:            REF BBLOCK;     ! Pointer to mechanism vector
;  1776          2807  2
;  1777          2808  2
;  1778          2809  2 ! If the signal is a failure status, unwind to the caller with that status as
;  1779          2810  2 ! the routine value.
;  1780          2811  2 !
;  1781          2812  2 IF .SIG[CHF$L_SIG_NAME] NEQ SS$_UNWIND AND NOT .SIG[CHF$L_SIG_NAME]
;  1782          2813  2 THEN
;  1783          2814  3     BEGIN
;  1784          2815  3     MCH[CHF$L_MCH_SAVR0] = .SIG[CHF$L_SIG_NAME];
;  1785          2816  3     $UNWIND();
;  1786          2817  2     END;
;  1787          2818  2
;  1788          2819  2
;  1789          2820  2 SS$_RESIGNAL
;  1790          2821  1 END;


                                                        .EXTRN  SYS$UNWIND

                                        0000 00000 INITIALIZE_QUEUE_FILE_HANDLER:
                                                        .WORD   Save nothing                    ; 2774
                    51        04    AC  D0 00002        MOVL    SIG, R1                         ; 2812
        00000920    8F        04    A1  D1 00006        CMPL    4(R1), #2336                    ;
```

```
                                    16  13 0000E          BEQL    1$                                      ;
                                12  04  A1  E8 00010      BLBS    4(R1), 1$                               ;
                                50  08  AC  D0 00014      MOVL    MCH, R0                                 ; 2815
                             0C AO  04  A1  D0 00018      MOVL    4(R1), 12(R0)                           ;
                                    7E  7C 0001D          CLRQ    -(SP)                                   ; 2816
                    00000000G   00      02  FB 0001F      CALLS   #2, SYS$UNWIND                          ;
                                50  0918 8F  3C 00026 1$:  MOVZWL  #2328, R0                              ; 2821
                                        04 0002B          RET                                             ;

; Routine Size:  44 bytes,    Routine Base:  CODE + 0DA4
```

```
;  1792          2822  1 GLOBAL ROUTINE INITIALIZE_QUEUE_FILE=
;  1793          2823  1
;  1794          2824  1 !++
;  1795          2825  1 !
;  1796          2826  1 ! FUNCTIONAL DESCRIPTION:
;  1797          2827  1 !       This routine opens and initializes the system job queue file.
;  1798          2828  1 !
;  1799          2829  1 ! INPUT PARAMETERS:
;  1800          2830  1 !       NONE
;  1801          2831  1 !
;  1802          2832  1 ! IMPLICIT INPUTS:
;  1803          2833  1 !       NONE
;  1804          2834  1 !
;  1805          2835  1 ! OUTPUT PARAMETERS:
;  1806          2836  1 !       NONE
;  1807          2837  1 !
;  1808          2838  1 ! IMPLICIT OUTPUTS:
;  1809          2839  1 !       NONE
;  1810          2840  1 !
;  1811          2841  1 ! ROUTINE VALUE:
;  1812          2842  1 !       Completion status.
;  1813          2843  1 !
;  1814          2844  1 ! SIDE EFFECTS:
;  1815          2845  1 !       NONE
;  1816          2846  1 !
;  1817          2847  1 !--
;  1818          2848  1
;  1819          2849  2 BEGIN
;  1820          2850  2 LOCAL
;  1821          2851  2         BUFFER:         BBLOCK[SYM$S_SYM],
;  1822          2852  2         DESC:           VECTOR[2],      ! Utility descriptor
;  1823          2853  2         RESNAM:         BBLOCK[30],     ! Buffer for resource name
;  1824          2854  2         STATUS_1,                       ! Status return
;  1825          2855  2         STATUS_2,                       ! Status return
;  1826          2856  2         STATUS_3;                       ! Status return
;  1827          2857  2 BUILTIN
;  1828          2858  2         FP;
;  1829          2859  2
;  1830          2860  2
;  1831          2861  2 ! Ensure that the queue reference count is zero, and establish the handler.
;  1832          2862  2 ! If this routine encounters any errors, the logic in UNLOCK_QUEUE_FILE will
;  1833          2863  2 ! ensure that all resources are properly released.
;  1834          2864  2 !
;  1835          2865  2 QUEUE_REFERENCE_COUNT = 0;
;  1836          2866  2 .FP = INITIALIZE_QUEUE_FILE_HANDLER;
;  1837          2867  2
;  1838          2868  2
;  1839          2869  2 ! Determine values for the initial file allocation (ALQ) and multibuffer count
;  1840          2870  2 ! (MBF) fields.
;  1841          2871  2 !
;  1842          2872  2 ! If the SJC$_EXTEND_QUANTITY item code is not present or specified as 0, then
;  1843          2873  2 ! use the job contoller default value. Likewise, if the SJC$_BUFFER_COUNT item
;  1844          2874  2 ! code is not present or specified as 0, then use the job controller default
;  1845          2875  2 ! value.
;  1846          2876  2 !
;  1847          2877  2 QUEUE_ALQ = .VALUE_EXTEND_QUANTITY;
;  1848          2878  2 IF .QUEUE_ALQ EQL 0 THEN QUEUE_ALQ = JBC$K_QUEUE_ALQ;
```

```
; 1849          2879  2 IF .QUEUE_ALQ LSSU 10 THEN QUEUE_ALQ = 10;
; 1850          2880  2 QUEUE_MBF = .VALUE_BUFFER_COUNT;
; 1851          2881  2 IF .QUEUE_MBF EQL 0 THEN QUEUE_MBF = JBC$K_QUEUE_MBF;
; 1852          2882  2
; 1853          2883  2
; 1854          2884  2 ! The following loop is executed once if the file is created or if it already
; 1855          2885  2 ! exists and has satisfactory attributes.  Otherwise it is executed twice to
; 1856          2886  2 ! unconditionally create the file.  However, if the /NEW_VERSION qualifier is
; 1857          2887  2 ! given, only execute the loop once to unconditionally create the file.
; 1858          2888  2 !
; 1859          2889  2 DECR I FROM (IF .ITEM_PRESENT[SJC$_NEW_VERSION] THEN 0 ELSE 1) TO 0 DO
; 1860          2890  3     BEGIN
; 1861        P 2891  3     $FAB_INIT(FAB=QUEUE_FAB,
; 1862        P 2892  3         ALQ=.QUEUE_ALQ,
; 1863        P 2893  3         DNA=UPLIT BYTE('SYS$SYSTEM:JBCSYSQUE.DAT'),
; 1864        P 2894  3         DNS=%CHARCOUNT('SYS$SYSTEM:JBCSYSQUE.DAT'),
; 1865        P 2895  3         FAC=<GET,PUT,UPD>,
; 1866        P 2896  3         FNA=.VALUE_QUEUE_FILE_SPECIFICATION[SDSC_A_POINTER],
; 1867        P 2897  3         FNS=.VALUE_QUEUE_FILE_SPECIFICATION[SDSC_W_LENGTH],
; 1868        P 2898  3         FOP=CBT,
; 1869        P 2899  3         GBC=JBC$K_QUEUE_GBC,
; 1870        P 2900  3         MRS=SYM$S_SYM,
; 1871        P 2901  3         NAM=QUEUE_NAM,
; 1872        P 2902  3         ORG=SEQ,
; 1873        P 2903  3         RFM=FIX,
; 1874        P 2904  3         SHR=<SHRGET,SHRPUT,SHRUPD>,
; 1875          2905  3         XAB=QUEUE_XAB);
; 1876          2906  3
; 1877        P 2907  3     $RAB_INIT(RAB=QUEUE_RAB,
; 1878        P 2908  3         FAB=QUEUE_FAB,
; 1879        P 2909  3         KSZ=4,
; 1880        P 2910  3         MBC=JBC$K_QUEUE_MBC,
; 1881        P 2911  3         MBF=.QUEUE_MBF,
; 1882        P 2912  3         RAC=KEY,
; 1883        P 2913  3         RSZ=SYM$S_SYM,
; 1884          2914  3         USZ=SYM$S_SYM);
; 1885          2915  3
; 1886        P 2916  3     $NAM_INIT(NAM=QUEUE_NAM,
; 1887        P 2917  3         ESA=QUEUE_RSA,
; 1888        P 2918  3         ESS=NAM$C_MAXRSS,
; 1889        P 2919  3         RSA=QUEUE_RSA,
; 1890          2920  3         RSS=NAM$C_MAXRSS);
; 1891          2921  3
; 1892        P 2922  3     $XABPRO_INIT(XAB=QUEUE_XAB,
; 1893        P 2923  3         PRO=<RWED,RWED,,>,
; 1894          2924  3         UIC=[1,4]);
; 1895          2925  3
; 1896          2926  3     QUEUE_FAB[FAB$V_CIF] = .I;
; 1897          2927  3
; 1898          2928  3
; 1899          2929  3     ! Create the queue file or open an existing queue file,
; 1900          2930  3     ! and connect the RAB.
; 1901          2931  3     !
; 1902          2932  3     IF $CREATE(FAB=QUEUE_FAB)
; 1903          2933  3     THEN
; 1904          2934  4         BEGIN
; 1905          2935  5         IF NOT $CONNECT(RAB=QUEUE_RAB)
```

```
;  1906          2936  4          THEN
;  1907          2937  4              RETURN .QUEUE_RAB[RAB$L_STS];
;  1908          2938  4          END
;  1909          2939  3      ELSE
;  1910          2940  4          BEGIN
;  1911          2941  4          IF .I EQL 0 THEN RETURN .QUEUE_FAB[FAB$L_STS];
;  1912          2942  4          QUEUE_FAB[FAB$W_MRS] = 0;
;  1913          2943  3          END;
;  1914          2944  3
;  1915          2945  3
;  1916          2946  3      ! Determine whether or not a new file was created.
;  1917          2947  3      !
;  1918          2948  3      IF .QUEUE_FAB[FAB$L_STS] EQL RMS$_CREATED OR NOT .QUEUE_FAB[FAB$V_CIF]
;  1919          2949  3      THEN FLAGS[FLAGS_V_QUEUE_CREATED] = TRUE
;  1920          2950  3      ELSE FLAGS[FLAGS_V_QUEUE_CREATED] = FALSE;
;  1921          2951  3
;  1922          2952  3
;  1923          2953  3      ! Check the file attributes and the first block of the file to ensure that
;  1924          2954  3      ! the file is suitable.
;  1925          2955  3      !
;  1926          2956  3      IF
;  1927          2957  4          BEGIN
;  1928          2958  4          IF .QUEUE_FAB[FAB$W_MRS] EQL SYM$S_SYM
;  1929          2959  4          AND .QUEUE_FAB[FAB$B_ORG] EQL FAB$C_SEQ
;  1930          2960  4          AND .QUEUE_FAB[FAB$B_RAT] EQL 0
;  1931          2961  4          AND .QUEUE_FAB[FAB$B_RFM] EQL FAB$C_FIX
;  1932          2962  4          THEN
;  1933          2963  4              IF .FLAGS[FLAGS_V_QUEUE_CREATED]
;  1934          2964  4              THEN
;  1935          2965  4                  TRUE
;  1936          2966  4              ELSE
;  1937          2967  5                  BEGIN
;  1938          2968  5                  QUEUE_RAB[RAB$L_KBF] = SQH_KBF;
;  1939          2969  5                  QUEUE_RAB[RAB$L_ROP] = RAB$M_NLK OR RAB$M_RRL;
;  1940          2970  5                  QUEUE_RAB[RAB$L_UBF] = BUFFER;
;  1941          2971  5                  DIAG_COUNT[0] = .DIAG_COUNT[0] + 1;
;  1942          2972  6                  IF $GET(RAB=QUEUE_RAB)
;  1943          2973  5                  THEN
;  1944          2974  5                      .BUFFER[SYM$B_TYPE] EQL SYM$K_SQH AND
;  1945          2975  5                      .BUFFER[SQH$W_STRUCTURE_LEVEL] EQL SQH$K_STRUCTURE_LEVEL
;  1946          2976  5                  ELSE
;  1947          2977  5                      FALSE
;  1948          2978  5                  END
;  1949          2979  4          ELSE
;  1950          2980  4              FALSE
;  1951          2981  4          END
;  1952          2982  3      THEN
;  1953          2983  3          EXITLOOP;
;  1954          2984  3
;  1955          2985  3
;  1956          2986  3      ! Close the file and loop to unconditionally create a new file. Display an
;  1957          2987  3      ! informational message stating that the file found has an incompatible
;  1958          2988  3      ! structure level.
;  1959          2989  3      !
;  1960          2990  4      IF NOT $CLOSE(FAB=QUEUE_FAB)
;  1961          2991  3      THEN RETURN .QUEUE_FAB[FAB$L_STS];
;  1962          2992  3      IF .BUFFER[SYM$B_TYPE] EQL SYM$K_SQH AND
```

```
;  1963     2993  3          .BUFFER[SQH$W_STRUCTURE_LEVEL] NEQ SQH$K_STRUCTURE_LEVEL
;  1964     2994  3      THEN
;  1965     2995  3          SIGNAL(JBC$_STRUCT_LEVEL OR STS$K_INFO,
;  1966     2996  3          1, .BUFFER[SQH$W_STRUCTURE_LEVEL]);
;  1967     2997  2      END;
;  1968     2998  2
;  1969     2999  2
;  1970     3000  2  ! Issue a message if the queue file was created.
;  1971     3001  2  !
;  1972     3002  2  IF .FLAGS[FLAGS_V_QUEUE_CREATED]
;  1973     3003  2  THEN
;  1974     3004  3      BEGIN
;  1975     3005  3      DESC[0] = .QUEUE_NAM[NAM$B_RSL];
;  1976     3006  3      DESC[1] = .QUEUE_NAM[NAM$L_RSA];
;  1977     3007  3      SIGNAL(JBC$_NEWQUEUE OR STS$K_INFO, 1, DESC);
;  1978     3008  2      END;
;  1979     3009  2
;  1980     3010  2
;  1981     3011  2  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
;  1982     3012  2  ! Disable the following code segment as currently the queue file must be opened
;  1983     3013  2  ! for shared access in order for RMS to support more than two local buffers.
;  1984     3014  2  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
;  1985     3015  2  WORK_AREA[0] = 1;                                ! Non-zero value disables option
;  1986     3016  2
;  1987     3017  2  ! First check to see if the use of a global buffer cache (FAB$W_GBC) has been
;  1988     3018  2  ! specified (to be used in place of a local buffer cache (RAB$B_MBF)). If global
;  1989     3019  2  ! buffers are to be used, then file sharing must remain enabled (FAB$B_SHR) and
;  1990     3020  2  ! the FLAGS_V_QUEUE_SHARED flag set. Otherwise, try to optimize the access.
;  1991     3021  2  !
;  1992     3022  2  ! The queue file has been opened optimized for a cluster environment, that is,
;  1993     3023  2  ! it is open for shared access. Next determine whether or not the queue file
;  1994     3024  2  ! really can be accessed by another job controller. If this node (1) is not
;  1995     3025  2  ! <or cannot become> a member of a cluster, or (2) the queue file is on a device
;  1996     3026  2  ! that is not cluster-wide accessible, then we should optimize for a non-cluster
;  1997     3027  2  ! environment. To do this, we'll close the queue file and reopen it with the
;  1998     3028  2  ! user-provided-interlocking (UPI) option selected. This will prevent RMS from
;  1999     3029  2  ! taking out locks on records in the local buffer cache which should reduce
;  2000     3030  2  ! locking overhead considerably in a non-cluster environment.
;  2001     3031  2  !
;  2002     3032  2  FLAGS[FLAGS_V_QUEUE_SHARED] = TRUE;
;  2003     3033  2
;  2004     3034  2  IF .QUEUE_FAB[FAB$W_GBC] EQL 0 AND .WORK_AREA[0] EQL 0
;  2005     3035  2  THEN
;  2006     3036  3      BEGIN
;  2007     3037  3      OWN
;  2008     3038  3          DEVCHAR2:               BBLOCK[4],      ! 2nd device char longword
;  2009     3039  3          CLUSTER_MEMBER:         BBLOCK[4];      ! Cluster member status
;  2010     3040  3
;  2011     3041  3      DESC[0] = .(QUEUE_NAM[NAM$T_DVI])<0,8>;     ! Form a string descriptor from
;  2012     3042  3      DESC[1] = QUEUE_NAM[NAM$T_DVI] + 1;         !  a counted string
;  2013   P 3043  3      STATUS_1 = $GETDVIW(
;  2014   P 3044  3          EFN=JBC$K_SYNC_EFN,
;  2015   P 3045  3          DEVNAM=DESC,
;  2016   P 3046  3          ITMLST=UPLIT(
;  2017   P 3047  3              WORD(4, DVI$_DEVCHAR2), LONG(DEVCHAR2, 0),
;  2018     3048  3              LONG(0)));
;  2019   P 3049  3      STATUS_2 = $GETSYIW(
```

```
;2020        P 3050  3          EFN=JBC$K_SYNC_EFN,
;2021        P 3051  3          ITMLST=UPLIT(
;2022        P 3052  3              WORD(1, SYI$_CLUSTER_MEMBER), LONG(CLUSTER_MEMBER, 0),
;2023          3053  3              LONG(0));
;2024          3054  3
;2025          3055  3      IF .STATUS_1 AND .STATUS_2
;2026          3056  3      THEN
;2027          3057  4          BEGIN
;2028          3058  5          IF (NOT .DEVCHAR2[DEV$V_CLU]) OR (NOT .CLUSTER_MEMBER<0,1>)
;2029          3059  4          THEN
;2030          3060  5              BEGIN
;2031          3061  5              FLAGS[FLAGS_V_QUEUE_SHARED] = FALSE;
;2032          3062  6              IF NOT $CLOSE(FAB=QUEUE_FAB)
;2033          3063  5                  THEN RETURN .QUEUE_FAB[FAB$L_STS];
;2034          3064  5              QUEUE_FAB[FAB$B_SHR] = FAB$M_SHRGET OR FAB$M_UPI;
;2035          3065  6              IF $OPEN(FAB=QUEUE_FAB)
;2036          3066  5              THEN
;2037          3067  6                  BEGIN
;2038          3068  7                  IF NOT $CONNECT(RAB=QUEUE_RAB)
;2039          3069  6                      THEN RETURN .QUEUE_RAB[RAB$L_STS];
;2040          3070  6                  END
;2041          3071  5              ELSE RETURN .QUEUE_FAB[FAB$L_STS];
;2042          3072  4              END;
;2043          3073  3          END;
;2044          3074  2 END;
;2045          3075  2
;2046          3076  2
;2047          3077  2 ! Take out the queue file master lock. The first job controller to access
;2048          3078  2 ! the queue file holds this lock, and is said to be the queue master.  If
;2049          3079  2 ! that job controller should fail, the next job controller to queue for the
;2050          3080  2 ! lock receives the completion AST, and can take necessary actions.
;2051          3081  2 !
;2052          3082  2 DESC[0] = 26;
;2053          3083  2 DESC[1] = RESNAM;
;2054          3084  2 RESNAM[0,0,32,0] = 'JBC$';
;2055          3085  2 CH$MOVE(22, QUEUE_NAM[NAM$T_DVI], RESNAM[4,0,0,0]);
;2056        P 3086  2 STATUS_1 = $ENQ(
;2057        P 3087  2     LKMODE=LCK$K_EXMODE,
;2058        P 3088  2     LKSB=QUEUE_FILE_LKSB,
;2059        P 3089  2     FLAGS=LCK$M_SYNCSTS OR LCK$M_NODLCKWT,
;2060        P 3090  2     RESNAM=DESC,
;2061          3091  2     ASTADR=QUEUE_MASTER_AST);
;2062          3092  2 IF NOT .STATUS_1 THEN RETURN .STATUS_1;
;2063          3093  2
;2064          3094  2
;2065          3095  2 ! Take out the queue lock.
;2066          3096  2 !
;2067          3097  2 DESC[0] = 30;
;2068          3098  2 RESNAM[26,0,32,0] = 'LOCK';
;2069        P 3099  2 STATUS_2 = $ENQW(
;2070        P 3100  2     EFN=JBC$K_SYNC_EFN,
;2071        P 3101  2     LKMODE=LCK$K_EXMODE,
;2072        P 3102  2     LKSB=QUEUE_LOCK_LKSB,
;2073          3103  2     RESNAM=DESC);
;2074          3104  2 IF .STATUS_2 THEN STATUS_2 = .QUEUE_LOCK_LKSB[0];
;2075          3105  2 IF NOT .STATUS_2 THEN RETURN .STATUS_2;
;2076          3106  2 FLAGS[FLAGS_V_QUEUE_LOCKED] = TRUE;
```

```
;  2077          3107  2
;  2078          3108  2
;  2079          3109  2 ! If this is not the first access to the file, do a brief reconstruction.
;  2080          3110  2 ! Otherwise, if the file was just created then initialize it. If not, then
;  2081          3111  2 ! this is the first access to an existing file, so do a full reconstruction.
;  2082          3112  2 !
;  2083          3113  2 IF .FLAGS[FLAGS_V_OMIT_QF_INIT]          ! For debugging purposes omit
;  2084          3114  2 THEN                                     !  initialization processing to examine
;  2085          3115  2     DIAG_TRACE[8] = -1                  !  a corrupted queue file on-line
;  2086          3116  2 ELSE
;  2087          3117  2     IF .STATUS_1<0,16> NEQ SS$_SYNCH
;  2088          3118  2     THEN
;  2089          3119  2         WARM_START_EXISTING_FILE(THIS_SYSID)
;  2090          3120  2     ELSE
;  2091          3121  2         IF .FLAGS[FLAGS_V_QUEUE_CREATED]
;  2092          3122  2         THEN COLD_START_NEW_FILE()
;  2093          3123  2         ELSE COLD_START_EXISTING_FILE();
;  2094          3124  2
;  2095          3125  2
;  2096          3126  2 ! Enable this job controller's remote request lock.
;  2097          3127  2 !
;  2098          3128  2 DESC[0] = 10;
;  2099          3129  2 DESC[1] = RESNAM;
;  2100          3130  2 RESNAM[0,0,32,0] = 'JBC$';
;  2101          3131  2 COPY_SYSID(THIS_SYSID, RESNAM[4,0,0,0]);
;  2102        P 3132  2 STATUS_3 = $ENQW(
;  2103        P 3133  2     EFN=JBC$K_SYNC_EFN,
;  2104        P 3134  2     LKMODE=LCK$K_EXMODE,
;  2105        P 3135  2     LKSB=REMOTE_REQUEST_LKSB,
;  2106        P 3136  2     FLAGS=LCK$M_NOQUEUE OR LCK$M_NODLCKBLK,
;  2107        P 3137  2     RESNAM=DESC,
;  2108          3138  2     BLKAST=REMOTE_BLOCKING_AST);
;  2109          3139  2 IF .STATUS_3 THEN STATUS_3 = .REMOTE_REQUEST_LKSB[0];
;  2110          3140  2 IF NOT .STATUS_3 AND .STATUS_3<0,16> NEQ SS$_NOTQUEUED THEN RETURN .STATUS_3;
;  2111          3141  2
;  2112          3142  2
;  2113          3143  2 ! Increment the queue's reference count to indicate a successful operation,
;  2114          3144  2 ! and return success.
;  2115          3145  2 !
;  2116          3146  2 QUEUE_REFERENCE_COUNT = 1;
;  2117          3147  2 SS$_NORMAL
;  2118          3148  1 END;


                                                                .PSECT  DATA,NOEXE,2

                                                    00000 DEVCHAR2:
                                                                .BLKB   4
                                                    00004 CLUSTER_MEMBER:
                                                                .BLKB   4

                                                                .PSECT  CODE,NOWRT,2

53  43  42  4A  3A  4D  45  54  53  59  53  24  53  59  53  00DD0 P.AAG:  .ASCII  \SYS$SYSTEM:JBCSYSQUE.DAT\
                        54  41  44  2E  45  55  51  53  59  00DDF
                                            00E6  0004  00DE8 P.AAH:  .WORD   4, 230
```

```
                                00000000'  00DEC          .ADDRESS DEVCHAR2                                       ;
                                00000000   00DF0          .LONG    0                                              ;
                                00000000   00DF4          .LONG    0                                              ;
                              10CF  0001   00DF8 P.AAI:    .WORD    1, 4303                                        ;
                                00000000'  00DFC          .ADDRESS CLUSTER_MEMBER                                 ;
                                00000000   00E00          .LONG    0                                              ;
                                00000000   00E04          .LONG    0                                              ;

                                                          $RMS_PTR=           QUEUE_FAB                           ;
                                                          $RMS_PTR=           QUEUE_RAB                           ;
                                                          $RMS_PTR=           QUEUE_NAM                           ;
                                                          $RMS_PTR=           QUEUE_XAB                           ;
                                                                   .EXTRN  SYS$CREATE, SYS$CONNECT                ;
                                                                   .EXTRN  SYS$GET, SYS$CLOSE                     ;
                                                                   .EXTRN  SYS$GETDVIW, SYS$GETSYIW               ;
                                                                   .EXTRN  SYS$OPEN, SYS$ENQ                      ;
                                                                   .EXTRN  SYS$ENQW                               ;

                                                    0FFC 00000     .ENTRY  INITIALIZE_QUEUE_FILE, Save R2,R3,R4,R5,R6,-; 2822
                                                                           R7,R8,R9,R10,R11                        ;
                                5B 00000000G   00    9E 00002     MOVAB   SYS$CLOSE, R11                         ;
                                5A 00000000G   00    9E 00009     MOVAB   SYS$CONNECT, R10                       ;
                                59       89    AF    9E 00010     MOVAB   INITIALIZE_QUEUE_FILE_HANDLER, R9      ;
                                58 00000000'   EF    9E 00014     MOVAB   FLAGS, R8                              ;
                                5E     FDD8    CE    9E 0001B     MOVAB   -552(SP), SP                           ;
                                         38    A8    D4 00020     CLRL    QUEUE_REFERENCE_COUNT                  ; 2865
                                6D             69    9E 00023     MOVAB   INITIALIZE_QUEUE_FILE_HANDLER, (FP)    ; 2866
                          0360  C8     0C8C    C8    3C 00026     MOVZWL  VALUE_EXTEND_QUANTITY, QUEUE_ALQ       ; 2877
                                               06    12 0002D     BNEQ    1$                                     ; 2878
                          0360  C8       64    8F    9A 0002F     MOVZBL  #100, QUEUE_ALQ                        ;
                                0A     0360    C8    D1 00035 1$:  CMPL    QUEUE_ALQ, #10                         ; 2879
                                               05    1E 0003A     BGEQU   2$                                     ;
                          0360  C8             0A    D0 0003C     MOVL    #10, QUEUE_ALQ                         ;
                          0364  C8     0C44    C8    90 00041 2$:  MOVB    VALUE_BUFFER_COUNT, QUEUE_MBF          ; 2880
                                               05    12 00048     BNEQ    3$                                     ; 2881
                          0364  C8             32    90 0004A     MOVB    #50, QUEUE_MBF                         ;
                                04     0C0D    C8    E9 0004F 3$:  BLBC    ITEM_PRESENT+13, 4$                    ; 2889
                                               56    D4 00054     CLRL    R6                                     ;
                                               03    11 00056     BRB     5$                                     ;
                                56             01    D0 00058 4$:  MOVL    #1, R6                                 ;
                                               56    D6 0005B 5$:  INCL    I                                      ; 2992
                                            01AC   31    0005D     BRW     15$                                    ;
0050  8F        00             6E             00    2C 00060 6$:  MOVC5   #0, (SP), #0, #80, $RMS_PTR            ; 2905
                                      0114     C8          00067                                                  ;
                          0114  C8     5003    8F    B0 0006A     MOVW    #20483, $RMS_PTR                       ;
                          0118  C8 00200000    8F    D0 00071     MOVL    #2097152, $RMS_PTR+4                   ;
                          0124  C8     0360    C8    D0 0007A     MOVL    QUEUE_ALQ, $RMS_PTR+16                 ;
                          012A  C8     0B0B    8F    B0 00081     MOVW    #2827, $RMS_PTR+22                     ;
                                      0131     C8    94 00088     CLRB    $RMS_PTR+29                            ;
                          0133  C8             01    90 0008C     MOVB    #1, $RMS_PTR+31                        ;
                          0138  C8     0208    C8    9E 00091     MOVAB   QUEUE_XAB, $RMS_PTR+36                 ;
                          013C  C8     01A8    C8    9E 00098     MOVAB   QUEUE_NAM, $RMS_PTR+40                 ;
                          0140  C8     1165    C8    D0 0009F     MOVL    VALUE_QUEUE_FILE_SPECIFICATION+2, -    ;
                                                                   $RMS_PTR+44                                    ;
                          0144  C8     FF1E    CF    9E 000A6     MOVAB   P.AAG, $RMS_PTR+48                     ;
                          0148  C8     1163    C8    90 000AD     MOVB    VALUE_QUEUE_FILE_SPECIFICATION, $RMS_PTR+52 ;
                          0149  C8             18    90 000B4     MOVB    #24, $RMS_PTR+53                       ;
```

```
                        014A  C8      0200  8F  B0 000B9          MOVW    #512, $RMS_PTR+54                       ;
                              015C    C8  B4 000C0              CLRW    $RMS_PTR+72                             ;
    0044  8F        00              6E      00  2C 000C4          MOVC5   #0, (SP), #0, #68, $RMS_PTR             ; 2914
                                    0164    C8   000CB                                                          ;
                        0164  C8      4401  8F  B0 000CE          MOVW    #17409, $RMS_PTR                        ;
                        0182  C8            01  90 000D5          MOVB    #1, $RMS_PTR+30                         ;
                        0184  C8  02000200  8F  D0 000DA          MOVL    #33554944, $RMS_PTR+32                  ;
                        0198  C8            04  90 000E3          MOVB    #4, $RMS_PTR+52                         ;
                        019A  C8      0364  C8  90 000E8          MOVB    QUEUE_MBF, $RMS_PTR+54                  ;
                        019B  C8            01  90 000EF          MOVB    #1, $RMS_PTR+55                         ;
                        01A0  C8      0114  C8  9E 000F4          MOVAB   QUEUE_FAB, $RMS_PTR+60                  ;
    0060  8F        00              6E      00  2C 000FB          MOVC5   #0, (SP), #0, #96, $RMS_PTR             ; 2920
                                    01A8    C8   00102                                                          ;
                        01A8  C8      6002  8F  B0 00105          MOVW    #24578, $RMS_PTR                        ;
                        01AA  C8            01  8E 0010C          MNEGB   #1, $RMS_PTR+2                          ;
                        01AC  C8      0260  C8  9E 00111          MOVAB   QUEUE_RSA, $RMS_PTR+4                   ;
                        01B2  C8            01  8E 00118          MNEGB   #1, $RMS_PTR+10                         ;
                        01B4  C8      0260  C8  9E 0011D          MOVAB   QUEUE_RSA, $RMS_PTR+12                  ;
    C058  8F        00              6E      00  2C 00124          MOVC5   #0, (SP), #0, #88, $RMS_PTR             ; 2924
                                    0208    C8   0012B                                                          ;
                        0208  C8      5813  8F  B0 0012E          MOVW    #22547, $RMS_PTR                        ;
                        0210  C8      FF00  8F  B0 00135          MOVW    #-256, $RMS_PTR+8                       ;
                        0214  C8  00010004  8F  D0 0013C          MOVL    #65540, $RMS_PTR+12                     ;
    011B  C8        01              01      56  F0 00145          INSV    I, #1, #1, QUEUE_FAB+7                  ; 2926
                                    0114    C8  9F 0014C          PUSHAB  QUEUE_FAB                               ; 2932
                00000000G   00              01  FB 00150          CALLS   #1, SYS$CREATE                          ;
                            0D              50  E9 00157          BLBC    R0, 7$                                  ;
                                    0164    C8  9F 0015A          PUSHAB  QUEUE_RAB                               ; 2935
                            6A              01  FB 0015E          CALLS   #1, SYS$CONNECT                         ;
                            0E              50  E8 00161          BLBS    R0, 10$                                 ;
                                        015F    31  00164          BRW     23$                                     ; 2937
                                          56    D5  00167  7$:      TSTL    I                                       ; 2941
                                          03    12  00169          BNEQ    9$                                      ;
                                        015E    31  0016B  8$:      BRW     24$                                     ;
                                014A      C8    B4  0016E  9$:      CLRW    QUEUE_FAB+54                            ; 2942
                00010619    8F          011C  C8  D1  00172  10$:    CMPL    QUEUE_FAB+8, #67097                     ; 2948
                                          06    13  00178          BEQL    11$                                     ;
    05          011B        C8                01  E0  0017D          BBS     #1, QUEUE_FAB+7, 12$                    ;
                            68                04  88  00183  11$:    BISB2   #4, FLAGS                               ; 2949
                                              03  11  00186          BRB     13$                                     ;
                            68                04  8A  00188  12$:    BICB2   #4, FLAGS                               ; 2950
                0200        8F          014A  C8  B1  0018B  13$:    CMPW    QUEUE_FAB+54, #512                      ; 2958
                                          4D    12  00192          BNEQ    14$                                     ;
                                    0131  C8    95  00194          TSTB    QUEUE_FAB+29                            ; 2959
                                          47    12  00198          BNEQ    14$                                     ;
                                    0132  C8    95  0019A          TSTB    QUEUE_FAB+30                            ; 2960
                                          41    12  0019E          BNEQ    14$                                     ;
                            01      0133  C8    91  001A0          CMPB    QUEUE_FAB+31, #1                        ; 2961
                                          3A    12  001A5          BNEQ    14$                                     ;
    6D                      68                02  E0  001A7          BBS     #2, FLAGS, 18$                          ; 2963
                0194        C8          F049  CF  9E  001AB          MOVAB   SQH_KBF, QUEUE_RAB+48                   ; 2968
                0168        C8  00100008  8F  D0  001B2          MOVL    #1048584, QUEUE_RAB+4                   ; 2969
                0188        C8          28    AE  9E  001BB          MOVAB   BUFFER, QUEUE_RAB+36                    ; 2970
                                        FE60  C8  D6  001C1          INCL    DIAG_COUNT                              ; 2971
                                    0164  C8    9F  001C5          PUSHAB  QUEUE_RAB                               ; 2972
                00000000G   00              01  FB  001C9          CALLS   #1, SYS$GET                             ;
                            0E              50  E9  001D0          BLBC    R0, 14$                                 ;
```

```
                        04      2C      AE  91 001D3          CMPB    BUFFER+4, #4                    ; 2974
                                        08  12 001D7          BNEQ    14$
                  0401  8F      34      AE  B1 001D9          CMPW    BUFFER+12, #1025                ; 2975
                                        33  13 001DF          BEQL    17$
                              0114      C8  9F 001E1 14$:     PUSHAB  QUEUE_FAB                       ; 2990
                        6B              01  FB 001E5          CALLS   #1, SYS$CLOSE
                        80              50  E9 001E8          BLBC    R0, 8$
                        04      2C      AE  91 001EB          CMPB    BUFFER+4, #4                    ; 2992
                                        1B  12 001EF          BNEQ    15$
                  0401  8F      34      AE  B1 001F1          CMPW    BUFFER+12, #1025                ; 2993
                                        13  13 001F7          BEQL    15$
                        7E      34      AE  3C 001F9          MOVZWL  BUFFER+12, -(SP)                ; 2996
                                        01  DD 001FD          PUSHL   #1                              ; 2995
                           000484AB     8F  DD 001FF          PUSHL   #296107
              00000000G   00              03  FB 00205          CALLS   #3, LIB$SIGNAL
                        02              56  F4 0020C 15$:     SOBGEQ  I, 16$                          ; 2889
                                        03  11 0020F          BRB     17$
                                    FE4C  31 00211 16$:     BRW     6$
                  1E    68              02  E1 00214 17$:     BBC     #2, FLAGS, 19$                  ; 3002
                     20 AE      01AB     C8  9A 00218 18$:     MOVZBL  QUEUE_NAM+3, DESC               ; 3005
                     24 AE      01AC     C8  D0 0021E          MOVL    QUEUE_NAM+4, DESC+4             ; 3006
                           20   AE       9F  00224          PUSHAB  DESC                            ; 3007
                                        01  DD 00227          PUSHL   #1
                           0004842B     8F  DD 00229          PUSHL   #295979
              00000000G   00              03  FB 0022F          CALLS   #3, LIB$SIGNAL
                  FEC4    C8              01  D0 00236 19$:     MOVL    #1, WORK_AREA                   ; 3015
                        68              08  88 0023B          BISB2   #8, FLAGS                       ; 3032
                              015C      C8  B5 0023E          TSTW    QUEUE_FAB+72                    ; 3034
                                        04  12 00242          BNEQ    20$
                              FEC4      C8  D5 00244          TSTL    WORK_AREA
                                        03  13 00248 20$:     BEQL    21$
                                    0085  31 0024A          BRW     25$
                     20 AE      01BC     C8  9A 0024D 21$:     MOVZBL  QUEUE_NAM+20, DESC              ; 3041
                     24 AE      01BD     C8  9E 00253          MOVAB   QUEUE_NAM+21, DESC+4            ; 3042
                                        7E  7C 00259          CLRQ    -(SP)                           ; 3048
                                        7E  7C 0025B          CLRQ    -(SP)
                              FD7F      CF  9F 0025D          PUSHAB  P.AAH
                                34      AE  9F 00261          PUSHAB  DESC
                        7E              01  7D 00264          MOVQ    #1, -(SP)
              00000000G   00              08  FB 00267          CALLS   #8, SYS$GETDVIW
                        57              50  D0 0026E          MOVL    R0, STATUS_1
                                        7E  7C 00271          CLRQ    -(SP)                           ; 3053
                                        7E  D4 00273          CLRL    -(SP)
                              FD77      CF  9F 00275          PUSHAB  P.AAI
                                        7E  7C 00279          CLRQ    -(SP)
                                        01  DD 0027B          PUSHL   #1
              0000U000G   00              07  FB 0027D          CALLS   #7, SYS$GETSYIW
                        56              50  D0 00284          MOVL    R0, STATUS_2
                        48              57  E9 00287          BLBC    STATUS_1, 25$                   ; 3055
                        45              56  E9 0028A          BLBC    STATUS_2, 25$
                        07 00000000'    EF  E9 0028D          BLBC    DEVCHAR2, 22$                   ; 3058
                        37 00000000'    EF  E8 00294          BLBS    CLUSTER_MEMBER, 25$
                        68              08  8A 0029B 22$:     BICB2   #8, FLAGS                       ; 3061
                              0114      C8  9F 0029E          PUSHAB  QUEUE_FAB                       ; 3062
                        6B              01  FB 002A2          CALLS   #1, SYS$CLOSE
                        24              50  E9 002A5          BLBC    R0, 24$
                  012B  C8        42    8F  90 002A8          MOVB    #66, QUEUE_FAB+23               ; 3064
```

```
                                 0114  C8  9F 002AE          PUSHAB  QUEUE_FAB                                  ; 3065
                    00000000G    00        01  FB 002B2          CALLS   #1, SYS$OPEN
                                 10        50  E9 002B9          BLBC    R0, 24$
                                 0164      C8  9F 002BC          PUSHAB  QUEUE_RAB                                  ; 3068
                                 6A        01  FB 002C0          CALLS   #1, SYS$CONNECT
                                 0C        50  E8 002C3          BLBS    R0, 25$
                                 50   016C  C8  D0 002C6 23$:     MOVL    QUEUE_RAB+8, R0                            ; 3069
                                           04  002CB            RET
                                 50   011C  C8  D0 002CC 24$:     MOVL    QUEUE_FAB+8, R0                            ; 3071
                                           04  002D1            RET
                           20    AE          1A  D0 002D2 25$:     MOVL    #26, DESC                                  ; 3082
                           24    AE          6E  9E 002D6          MOVAB   RESNAM, DESC+4                             ; 3083
                                 6E 2443424A 8F  D0 002DA          MOVL    #608387658, RESNAM                         ; 3084
                 04  AE   01BC  C8          16  28 002E1          MOVC3   #22, QUEUE_NAM+20, RESNAM+4                ; 3085
                                             7E  7C 002E8          CLRQ    -(SP)                                      ; 3091
                                             7E  7C 002EA          CLRQ    -(SP)
                                 00000000G   EF  9F 002EC          PUSHAB  QUEUE_MASTER_AST
                                             7E  D4 002F2          CLRL    -(SP)
                                       38    AE  9F 002F4          PUSHAB  DESC
                                 7E    0208  8F  3C 002F7          MOVZWL  #520, -(SP)
                                       00A8  C8  9F 002FC          PUSHAB  QUEUE_FILE_LKSB
                                             05  DD 00300          PUSHL   #5
                                             7E  D4 00302          CLRL    -(SP)
                    00000000G    00          0B  FB 00304          CALLS   #11, SYS$ENQ
                                 57          50  D0 0030B          MOVL    R0, STATUS_1
                                 04          57  E8 0030E          BLBS    STATUS_1, 26$                              ; 3092
                                 50          57  D0 00311          MOVL    STATUS_1, R0
                                             04  00314            RET
                           20    AE          1E  D0 00315 26$:     MOVL    #30, DESC                                  ; 3097
                           1A    AE 4B434F4C 8F  D0 00319          MOVL    #1262702412, RESNAM+26                     ; 3098
                                             7E  7C 00321          CLRQ    -(SP)                                      ; 3103
                                             7E  7C 00323          CLRQ    -(SP)
                                             7E  7C 00325          CLRQ    -(SP)
                                       38    AE  9F 00327          PUSHAB  DESC
                                             7E  D4 0032A          CLRL    -(SP)
                                       00B0  C8  9F 0032C          PUSHAB  QUEUE_LOCK_LKSB
                                             05  DD 00330          PUSHL   #5
                                             01  DD 00332          PUSHL   #1
                    00000000G    00          0B  FB 00334          CALLS   #11, SYS$ENQW
                                 56          50  D0 0033B          MOVL    R0, STATUS_2
                                 08          56  E9 0033E          BLBC    STATUS_2, 27$                              ; 3104
                                 56   00B0   C8  3C 00341          MOVZWL  QUEUE_LOCK_LKSB, STATUS_2
                                 04          56  E8 00346          BLBS    STATUS_2, 28$                              ; 3105
                                 50          56  D0 00349 27$:     MOVL    STATUS_2, R0
                                             04  0034C            RET
                                 68          01  88 0034D 28$:     BISB2   #1, FLAGS                                  ; 3106
                 07        02    A8          06  E1 00350          BBC     #6, FLAGS+2, 29$                           ; 3113
                        FE20     C8          01  CE 00355          MNEGL   #1, DIAG_TRACE+32                          ; 3115
                                             21  11 0035A          BRB     32$
                        0689     8F          57  B1 0035C 29$:     CMPW    STATUS_1, #1673                            ; 3117
                                             0A  13 00361          BEQL    30$
                                       08    A8  9F 00363          PUSHAB  THIS_SYSID                                 ; 3119
                        FD9F     C9          01  FB 00366          CALLS   #1, @ARM_START_EXISTING_FILE
                                             10  11 0036B          BRB     32$
                 07                  68          02  E1 0036D 30$:     BBC     #2, FLAGS, 31$                         ; 3121
                        FC3E     C9          00  FB 00371          CALLS   #0, COLD_START_NEW_FILE                    ; 3122
                                             05  11 00376          BRB     32$
```

```
                F8C5   C9              00  FB 00378 31$:    CALLS   #0, COLD_START_EXISTING_FILE                ; 3123
                  20   AE              0A  D0 0037D 32$:    MOVL    #10, DESC                                 ; 3128
                  24   AE              6E  9E 00381         MOVAB   RESNAM, DESC+4                            ; 3129
                       6E 2443424A     8F  D0 00385         MOVL    #608387658, RESNAM                        ; 3130
                  04   AE        08    A8  D0 0038C         MOVL    THIS_SYSID, RESNAM+4                      ; 3131
                  08   AE        0C    A8  B0 00391         MOVW    THIS_SYSID+4, RESNAM+8
                                       7E  7C 00396         CLRQ    -(SP)                                     ; 3138
                           00000000G   EF  9F 00398         PUSHAB  REMOTE_BLOCKING_AST
                                       7E  7C 0039E         CLRQ    -(SP)
                                       7E  D4 003A0         CLRL    -(SP)
                                 38    AE  9F 003A2         PUSHAB  DESC
                           7E    0404  8F  3C 003A5         MOVZWL  #1028, -(SP)
                                 00A0  C8  9F 003AA         PUSHAB  REMOTE_REQUEST_LKSB
                                       05  DD 003AE         PUSHL   #5
                                       01  DD 003B0         PUSHL   #1
            00000000G  00              0B  FB 003B2         CALLS   #11, SYS$ENQW
                       08              50  E9 003B9         BLBC    STATUS_3, 33$                             ; 3139
                       50    00A0      C8  3C 003BC         MOVZWL  REMOTE_REQUEST_LKSB, STATUS_3
                       07              50  E8 003C1         BLBS    STATUS_3, 34$                             ; 3140
                  09B8 8F              50  B1 003C4 33$:    CMPW    STATUS_3, #2488
                                       07  12 003C9         BNEQ    35$
                  38   A8              01  D0 003CB 34$:    MCVL    #1, QUEUE_REFERENCE_COUNT                 ; 3146
                       50              01  D0 003CF         MOVL    #1, R0                                    ; 3148
                                           04 003D2 35$:    RET

; Routine Size:  979 bytes,    Routine Base:  CODE + 0E08
```

```
; 2120          3149  1 END
; 2121          3150  0 ELUDOM


                                                        .EXTRN  LIB$SIGNAL

;                               PSECT SUMMARY
;
;       Name                    Bytes                   Attributes
;
;  COMMON                         5024  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  OVR,NOPIC,ALIGN(2)
;  CODE                           4571  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  DATA                              8  NOVEC,  WRT,  RD ,NOEXE,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       183         0      1000          00:01.4


; Information:  1
; Warnings:     0
; Errors:       0


;                                       COMMAND QUALIFIERS

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

; Size:          4464 code + 5139 data bytes
; Run Time:         01:03.7
; Elapsed Time:     03:49.1
; Line/CPU Min:      2967
; Lexemes/CPU-Min: 40328
; Memory Used:  574 pages
; Compilation Complete
```

0193 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

JBCMSG
LIS

RECORDUTL
LIS

QUEUEUTIL
LIS