G 12

```
            JJJ     000000000     BBBBBBBBBBBB        CCCCCCCCCCC   TTTTTTTTTTTTTTT  LLL
            JJJ     000000000     BBBBBBBBBBBB        CCCCCCCCCCC   TTTTTTTTTTTTTTT  LLL
            JJJ     000000000     BBBBBBBBBBBB        CCCCCCCCCCC   TTTTTTTTTTTTTTT  LLL
            JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
            JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
            JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
            JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
            JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
            JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
            JJJ  000         000  BBBBBBBBBBBB     CCC                    TTT        LLL
            JJJ  000         000  BBBBBBBBBBBB     CCC                    TTT        LLL
            JJJ  000         000  BBBBBBBBBBBB     CCC                    TTT        LLL
JJJ         JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
JJJ         JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
JJJ         JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
JJJ         JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
JJJ         JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
JJJ         JJJ  000         000  BBB         BBB  CCC                    TTT        LLL
   JJJJJJJJJ        000000000     BBBBBBBBBBBB        CCCCCCCCCCC         TTT        LLLLLLLLLLLLLLL
   JJJJJJJJJ        000000000     BBBBBBBBBBBB        CCCCCCCCCCC         TTT        LLLLLLLLLLLLLLL
   JJJJJJJJJ        000000000     BBBBBBBBBBBB        CCCCCCCCCCC         TTT        LLLLLLLLLLLLLLL
```

```
_$2

Val
----
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
0000
314
485
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
7FF
```

```
**FILE**ID**CHECKPROT                                  L 13
```

```
  CCCCCCCC  HH      HH  EEEEEEEEEE    CCCCCCCC  KK      KK  PPPPPPPP    RRRRRRRR      OOOOOO    TTTTTTTTTT
  CCCCCCCC  HH      HH  EEEEEEEEEE    CCCCCCCC  KK      KK  PPPPPPPP    RRRRRRRR      OOOOOO    TTTTTTTTTT
CC          HH      HH  EE          CC          KK      KK  PP      PP  RR      RR  OO      OO      TT
CC          HH      HH  EE          CC          KK      KK  PP      PP  RR      RR  OO      OO      TT
CC          HH      HH  EE          CC          KK    KK    PP      PP  RR      RR  OO      OO      TT
CC          HH      HH  EE          CC          KK    KK    PP      PP  RR      RR  OO      OO      TT
CC          HHHHHHHHHH  EEEEEEEE    CC          KKKKKK      PPPPPPPP    RRRRRRRR    OO      OO      TT
CC          HHHHHHHHHH  EEEEEEEE    CC          KKKKKK      PPPPPPPP    RRRRRRRR    OO      OO      TT
CC          HH      HH  EE          CC          KK    KK    PP          RR  RR      OO      OO      TT
CC          HH      HH  EE          CC          KK    KK    PP          RR  RR      OO      OO      TT
CC          HH      HH  EE          CC          KK      KK  PP          RR    RR    OO      OO      TT          ....
CC          HH      HH  EE          CC          KK      KK  PP          RR    RR    OO      OO      TT          ....
  CCCCCCCC  HH      HH  EEEEEEEEEE    CCCCCCCC  KK      KK  PP          RR      RR    OOOOOO        TT          ....
  CCCCCCCC  HH      HH  EEEEEEEEEE    CCCCCCCC  KK      KK  PP          RR      RR    OOOOOO        TT          ....

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

```
0 MODULE CHECKPROT(%TITLE 'Queue protection checks'
0                 IDENT = 'V04-000'
0                 ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPCNSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:
1 !       Job controller.
1 !
1 ! ABSTRACT:
1 !       This module contains the routines that interface to the $CHKPRO service
1 !       for queue protection checking.
1 !
1 ! ENVIRONMENT:
1 !       VAX/VMS user and kernel mode.
1 !--
1 !
1 ! AUTHOR: M. Jack, CREATION DATE: 16-Feb-1982
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-001 MLJ0114         Martin L. Jack, 23-Jun-1983  4:57
1 !               Changes for job controller baselevel.
1 !
1 !**
```

```
;    52         0051   1 REQUIRE 'SRC$:JOBCTLDEF';
;    53         1092   1
;    54         1093   1
;    55         1094   1 FORWARD ROUTINE
;    56         1095   1         CHKPRO,
;    57         1096   1         CHECK_R_ACCESS,
;    58         1097   1         CHECK_W_ACCESS,
;    59         1098   1         CHECK_E_ACCESS,
;    60         1099   1         CHECK_D_ACCESS;
```

```
1 GLOBAL ROUTINE CHKPRO(ACCESS,OWNER,PROTECTION)=
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine executes a call to the $CHKPRO service.
1 !
1 ! INPUT PARAMETERS:
1 !       ACCESS          - Desired access.
1 !       OWNER           - Object ownership.
1 !       PROTECTION      - Object protection.
1 !
1 ! IMPLICIT INPUTS:
1 !       MBX             - Pointer to buffered mailbox message.
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       True if access is granted, false if denied.
1 !
1 ! SIDE EFFECTS:
1 !       NONE
1 !
1 !--
1
2 BEGIN
2 LOCAL
2       ITMLST:         BBLOCK[64],     ! $CHKPRO item list
2       RIGHTS:         VECTOR[2];      ! Rights vector
2
2
2 ! OPER privilege implicitly grants all types of access.
2 !
2 IF .BBLOCK[MBX[ACM$Q_PRVMSK], PRV$V_OPER]
2 THEN
2     RETURN TRUE;
2
2
2 ! Construct the item list and call $CHKPRO to check the access.
2 !
2 RIGHTS[0] = .MBX[ACM$L_UIC];
2 RIGHTS[1] = 0;
2 ITMLST[0,0,16,0] = 4;                            ! Desired access
2 ITMLST[2,0,16,0] = CHP$_ACCESS;
2 ITMLST[4,0,32,0] = ACCESS;
2 ITMLST[8,0,32,0] = 0;
2 ITMLST[12,0,16,0] = 8;                           ! Requestor's privileges
2 ITMLST[14,0,16,0] = CHP$_PRIV;
2 ITMLST[16,0,32,0] = MBX[ACM$Q_PRVMSK];
2 ITMLST[20,0,32,0] = 0;
2 ITMLST[24,0,16,0] = 8;                           ! Requestor's UIC
2 ITMLST[26,0,16,0] = CHP$_RIGHTS;
2 ITMLST[28,0,32,0] = RIGHTS;
```

```
:  119          1157  2 ITMLST[32,0,32,0] = 0;
:  120          1158  2 ITMLST[36,0,16,0] = 4;                      ! Object's UIC
:  121          1159  2 ITMLST[38,0,16,0] = CHP$_OWNER;
:  122          1160  2 ITMLST[40,0,32,0] = OWNER;
:  123          1161  2 ITMLST[44,0,32,0] = 0;
:  124          1162  2 ITMLST[48,0,16,0] = 4;                      ! Object's protection
:  125          1163  2 ITMLST[50,0,16,0] = CHP$_PROT;
:  126          1164  2 ITMLST[52,0,32,0] = PROTECTION;
:  127          1165  2 ITMLST[56,0,32,0] = 0;
:  128          1166  2 ITMLST[60,0,32,0] = 0;                      ! Trailing zero
:  129          1167  2
:  130          1168  2
:  131          1169  3 $CHKPRO(ITMLST=ITMLST)
:  132          1170  1 END;


                                                        .TITLE  CHECKPROT Queue protection checks
                                                        .IDENT  \V04-000\

                                                        .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
```

```
00023C 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   6
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   10
012F4 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

                                                     JBC$_CLOSEOUT=      266328
                                                     JBC$_NOCMKRNL=      272388
                                                     JBC$_NOOPER=        272532
                                                     JBC$_NOSYSNAM=      272404
                                                     JBC$_OPENIN=        266392
                                                     JBC$_OPENOUT=       266400
                                                     JBC$_READERR=       266416
                                                     JBC$_WRITEERR=      266448
                                                               .EXTRN  SYS$CHKPRO

                                                               .PSECT  CODE,NOWRT,2

                                         0000 00000            .ENTRY  CHKPRO, Save nothing                ; 1100
                  5E        B8      AE     9E 00002            MOVAB   -72(SP), SP                         ;
                  50 00000000'      EF     D0 00006            MOVL    MBX, R0                             ; 1137
    04        06  A0                02     E1 0000D            BBC     #2, 6(R0), 1$                       ;
                  50                01     D0 00012            MOVL    #1, R0                              ; 1139
                                       04 00015                RET                                         ;
                  6E        0C      A0     D0 00016 1$:        MOVL    12(R0), RIGHTS                      ; 1144
                            04      AE     D4 0001A            CLRL    RIGHTS+4                            ; 1145
              08  AE 00010004       8F     D0 0001D            MOVL    #65540, ITMLST                      ; 1146
              0C  AE        04      AC     9E 00025            MOVAB   ACCESS, ITMLST+4                    ; 1148
                            10      AE     D4 0002A            CLRL    ITMLST+8                            ; 1149
              14  AE 00030008       8F     D0 0002D            MOVL    #196616, ITMLST+12                  ; 1150
              18  AE        04      A0     9E 00035            MOVAB   4(R0), ITMLST+16                    ; 1152
                            1C      AE     D4 0003A            CLRL    ITMLST+20                           ; 1153
              20  AE 00060008       8F     D0 0003D            MOVL    #393224, ITMLST+24                  ; 1154
              24  AE                6E     9E 00045            MOVAB   RIGHTS, ITMLST+28                   ; 1156
                            28      AE     D4 00049            CLRL    ITMLST+32                           ; 1157
              2C  AE 000C0004       8F     D0 0004C            MOVL    #786436, ITMLST+36                  ; 1158
              30  AE        08      AC     9E 00054            MOVAB   OWNER, ITMLST+40                    ; 1160
                            34      AE     D4 00059            CLRL    ITMLST+44                           ; 1161
              38  AE 000D0004       8F     D0 0005C            MOVL    #851972, ITMLST+48                  ; 1162
```

```
                          3C   AE       0C   AC  9E 00064        MOVAB    PROTECTION, ITMLST+52                    ; 1164
                                        40   AE  7C 00069        CLRQ     ITMLST+56                                ; 1165
                                        08   AE  9F 0006C        PUSHAB   ITMLST                                   ; 1169
                   00000000G  00             01  FB 0006F        CALLS    #1, SYS$CHKPRO                           ;
                                                 04 00076        RET                                               ; 1170

; Routine Size:  119 bytes,    Routine Base:  CODE + 0000
```

```
1 GLOBAL ROUTINE CHECK_R_ACCESS(SMQ,SJH)=
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine checks whether the requestor has R access to the specified
1 !       job in the specified queue.  OPER privilege and E access to the queue
1 !       implicitly grant R access.
1 !
1 ! INPUT PARAMETERS:
1 !       SMQ             - Pointer to SMQ.
1 !       SJH             - Pointer to SJH.
1 !
1 ! IMPLICIT INPUTS:
1 !       MBX             - Pointer to buffered mailbox message.
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       True if access is granted, false if denied.
1 !
1 ! SIDE EFFECTS:
1 !       NONE
1 !
1 !--
1
2 BEGIN
2 MAP
2       SMQ:            REF BBLOCK,     ! Pointer to SMQ
2       SJH:            REF BBLOCK;     ! Pointer to SJH
2
2
2 ! Test for R access to the job.
2 !
2 IF CHKPRO(%B'0001', .SJH[SJH$L_UIC], .SMQ[SMQ$W_PROTECTION])
2 THEN
2     RETURN TRUE;
2
2
2 ! Test for E access to the queue.
2 !
2 IF CHKPRO(%B'0100', .SMQ[SMQ$L_OWNER_UIC], .SMQ[SMQ$W_PROTECTION])
2 THEN
2     RETURN TRUE;
2
2
2 FALSE
1 END;
```

```
                                          0004 00000          .ENTRY   CHECK_R_ACCESS, Save R2                      ; 1171
                    52         04     AC   D0 00002          MOVL     SMQ, R2                                      ; 1209
                    7E       0104     C2   3C 00006          MOVZWL   260(R2), -(SP)                               ;
                    50         08     AC   D0 0000B          MOVL     SJH, R0                                      ;
                             0144     C0   DD 0000F          PUSHL    324(R0)                                      ;
                                      01   DD 00013          PUSHL    #1                                           ;
             FF6F   CF                03   FB 00015          CALLS    #3, CHKPRO                                   ;
                    13                50   E8 0001A          BLBS     R0, 1$                                       ;
                    7E       0104     C2   3C 0001D          MOVZWL   260(R2), -(SP)                               ; 1216
                             00D0     C2   DD 00022          PUSHL    208(R2)                                      ;
                                      04   DD 00026          PUSHL    #4                                           ;
             FF5C   CF                03   FB 00028          CALLS    #3, CHKPRO                                   ;
                    04                50   E9 0002D          BLBC     R0, 2$                                       ;
                    50                01   D0 00030 1$:      MOVL     #1, R0                                       ; 1218
                                           04 00033          RET                                                   ;
                                      50   D4 00034 2$:      CLRL     R0                                           ; 1222
                                           04 00036          RET                                                   ;

; Routine Size:  55 bytes,    Routine Base:  CODE + 0077
;  187        1223  1 GLOBAL ROUTINE CHECK_W_ACCESS(SMQ)=
;  188        1224  1
;  189        1225  1 !++
;  190        1226  1 !
;  191        1227  1 ! FUNCTIONAL DESCRIPTION:
;  192        1228  1 !       This routine checks whether the requestor has W access to the specified
;  193        1229  1 !       queue.  OPER privilege and E access to the queue implicitly grant
;  194        1230  1 !       W access.
;  195        1231  1 !
;  196        1232  1 ! INPUT PARAMETERS:
;  197        1233  1 !       SMQ             - Pointer to SMQ.
;  198        1234  1 !
;  199        1235  1 ! IMPLICIT INPUTS:
;  200        1236  1 !       MBX             - Pointer to buffered mailbox message.
;  201        1237  1 !
;  202        1238  1 ! OUTPUT PARAMETERS:
;  203        1239  1 !       NONE
;  204        1240  1 !
;  205        1241  1 ! IMPLICIT OUTPUTS:
;  206        1242  1 !       NONE
;  207        1243  1 !
;  208        1244  1 ! ROUTINE VALUE:
;  209        1245  1 !       True if access is granted, false if denied.
;  210        1246  1 !
;  211        1247  1 ! SIDE EFFECTS:
;  212        1248  1 !       NONE
;  213        1249  1 !
;  214        1250  1 !--
;  215        1251  1
;  216        1252  2 BEGIN
;  217        1253  2 MAP
;  218        1254  2       SMQ:            REF BBLOCK;     ! Pointer to SMQ
;  219        1255  2
;  220        1256  2
;  221        1257  2 ! Test for W access to the queue.
;  222        1258  2 !
;  223        1259  2 IF CHKPRO(%B'0010', .SMQ[SMQ$L_OWNER_UIC], .SMQ[SMQ$W_PROTECTION])
;  224        1260  2 THEN
;  225        1261  2     RETURN TRUE;
;  226        1262  2
;  227        1263  2
;  228        1264  2 ! Test for E access to the queue.
;  229        1265  2 !
;  230        1266  2 IF CHKPRO(%B'0100', .SMQ[SMQ$L_OWNER_UIC], .SMQ[SMQ$W_PROTECTION])
;  231        1267  2 THEN
;  232        1268  2     RETURN TRUE;
;  233        1269  2
;  234        1270  2
;  235        1271  2 FALSE
;  236        1272  1 END;



                                        0004 00000        .ENTRY  CHECK_W_ACCESS, Save R2                 ; 1223
                       52       04   AC  D0 00002        MOVL    SMQ, R2                                 ; 1259
```

```
                 7E      0104    C2  3C 00006          MOVZWL  260(R2), -(SP)                     ;
                         00D0    C2  DD 0000B          PUSHL   208(R2)                            ;
                                 02  DD 0000F          PUSHL   #2                                 ;
          FF3C   CF              03  FB 00011          CALLS   #3, CHKPRO                         ;
                 13              50  E8 00016          BLBS    R0, 1$                             ;
                 7E      0104    C2  3C 00019          MOVZWL  260(R2), -(SP)                     ; 1266
                         00D0    C2  DD 0001E          PUSHL   208(R2)                            ;
                                 04  DD 00022          PUSHL   #4                                 ;
          FF29   CF              03  FB 00024          CALLS   #3, CHKPRO                         ;
                 04              50  E9 00029          BLBC    R0, 2$                             ;
                 50              01  D0 0002C  1$:     MOVL    #1, R0                             ; 1268
                                     04 0002F          RET                                        ;
                                 50  D4 00030  2$:     CLRL    R0                                 ; 1272
                                     04 00032          RET                                        ;

; Routine Size:  51 bytes,    Routine Base:  CODE + 00AE
;  238         1273   1 GLOBAL ROUTINE CHECK_E_ACCESS(SMQ)=
;  239         1274   1
;  240         1275   1 !++
;  241         1276   1 !
;  242         1277   1 ! FUNCTIONAL DESCRIPTION:
;  243         1278   1 !       This routine checks whether the requestor has E access to the specified
;  244         1279   1 !       queue.  OPER privilege implicitly grants E access.
;  245         1280   1 !
;  246         1281   1 ! INPUT PARAMETERS:
;  247         1282   1 !       SMQ              - Pointer to SMQ.
;  248         1283   1 !
;  249         1284   1 ! IMPLICIT INPUTS:
;  250         1285   1 !       MBX              - Pointer to buffered mailbox message.
;  251         1286   1 !
;  252         1287   1 ! OUTPUT PARAMETERS:
;  253         1288   1 !       NONE
;  254         1289   1 !
;  255         1290   1 ! IMPLICIT OUTPUTS:
;  256         1291   1 !       NONE
;  257         1292   1 !
;  258         1293   1 ! ROUTINE VALUE:
;  259         1294   1 !       True if access is granted, false if denied.
;  260         1295   1 !
;  261         1296   1 ! SIDE EFFECTS:
;  262         1297   1 !       NONE
;  263         1298   1 !
;  264         1299   1 !--
;  265         1300   1
;  266         1301   2 BEGIN
;  267         1302   2 MAP
;  268         1303   2       SMQ:             REF BBLOCK;     ! Pointer to SMQ
;  269         1304   2
;  270         1305   2
;  271         1306   2 ! Test for E access to the queue.
;  272         1307   2 !
;  273         1308   2 IF CHKPRO(%B'0100', .SMQ[SMQ$L_OWNER_UIC], .SMQ[SMQ$W_PROTECTION])
;  274         1309   2 THEN
;  275         1310   2     RETURN TRUE;
;  276         1311   2
;  277         1312   2
;  278         1313   2 FALSE
;  279         1314   1 END;



                                      0000 00000         .ENTRY  CHECK_E_ACCESS, Save nothing           ; 1273
                  50        04     AC   D0 00002         MOVL    SMQ, R0                                 ; 1308
                  7E      0104     C0   3C 00006         MOVZWL  260(R0), -(SP)                          ;
                          00D0     C0   DD 0000B         PUSHL   208(R0)                                 ;
                                   04   DD 0000F         PUSHL   #4                                      ;
          FF09    CF               03   FB 00011         CALLS   #3, CHKPRO                              ;
                  04               50   E9 00016         BLBC    R0, 1$                                  ;
                  50               01   D0 00019         MOVL    #1, R0                                  ; 1310
                                        04 0001C         RET                                             ;
                                   50   D4 0001D 1$:     CLRL    R0                                      ; 1314
```

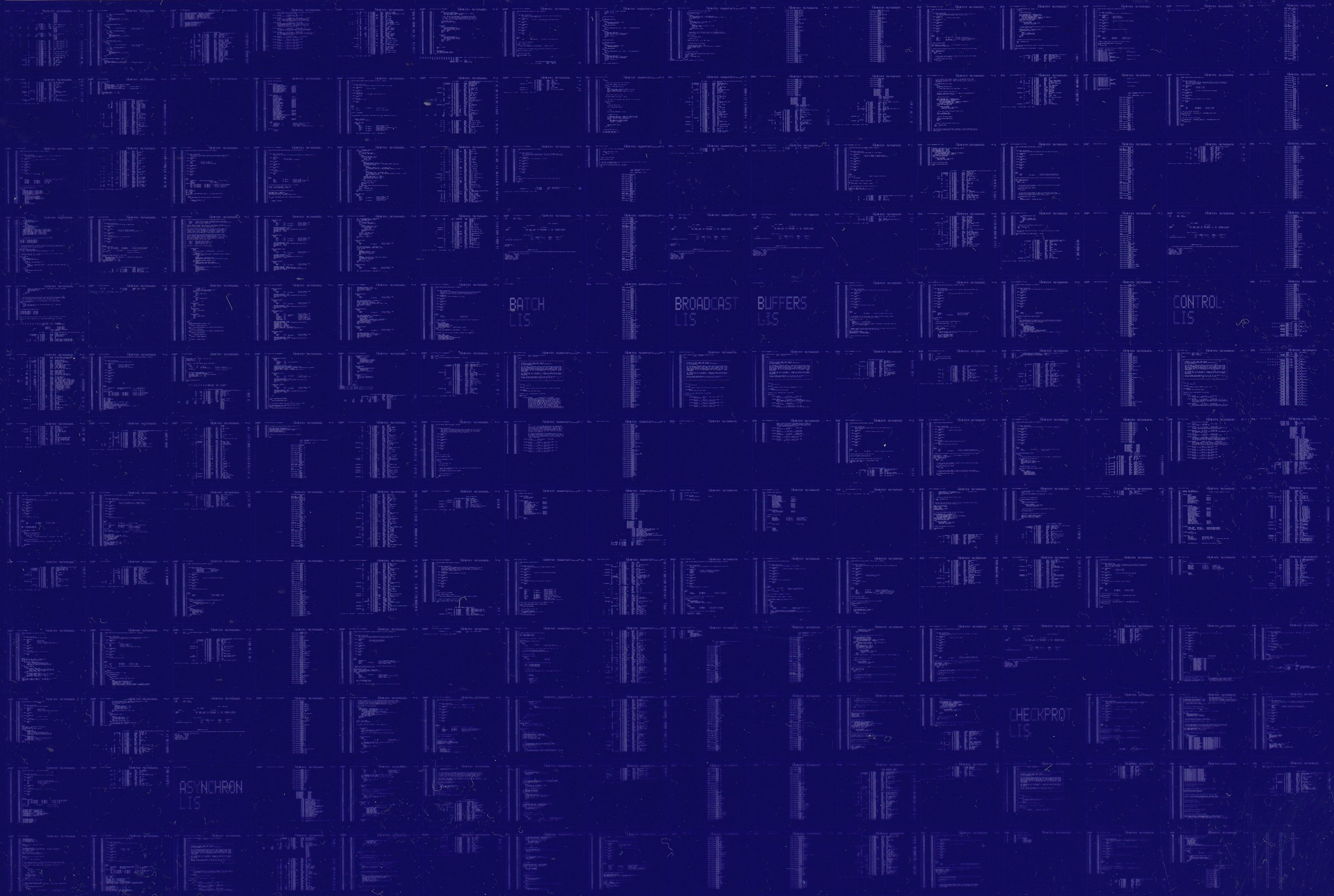
```
                                                  04 0001F          RET                                                 ;

; Routine Size:  32 bytes,    Routine Base:  CODE + 00E1
```

```
1 GLOBAL ROUTINE CHECK_D_ACCESS(SMQ,SJH)=
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine checks whether the requestor has D access to the specified
1 !       job in the specified queue.  OPER privilege and E access to the queue
1 !       implicitly grant D access.
1 !
1 ! INPUT PARAMETERS:
1 !       SMQ             - Pointer to SMQ.
1 !       SJH             - Pointer to SJH.
1 !
1 ! IMPLICIT INPUTS:
1 !       MBX             - Pointer to buffered mailbox message.
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       True if access is granted, false if denied.
1 !
1 ! SIDE EFFECTS:
1 !       NONE
1 !
1 !--
1
2 BEGIN
2 MAP
2       SMQ:            REF BBLOCK,     ! Pointer to SMQ
2       SJH:            REF BBLOCK;     ! Pointer to SJH
2
2
2 ! Test for D access to the job.
2 !
2 IF CHKPRO(%B'1000', .SJH[SJH$L_UIC], .SMQ[SMQ$W_PROTECTION])
2 THEN
2     RETURN TRUE;
2
2
2 ! Test for E access to the queue.
2 !
2 IF CHKPRO(%B'0100', .SMQ[SMQ$L_OWNER_UIC], .SMQ[SMQ$W_PROTECTION])
2 THEN
2     RETURN TRUE;
2
2
2 FALSE
1 END;
```

```
                                  0004 00000          .ENTRY  CHECK_D_ACCESS, Save R2                    ; 1315
                  52       04   AC  D0 00002          MOVL    SMQ, R2                                    ; 1353
                  7E     0104   C2  3C 00006          MOVZWL  260(R2), -(SP)                             ;
                  50       08   AC  D0 0000B          MOVL    SJH, R0                                    ;
                         0144   C0  DD 0000F          PUSHL   324(R0)                                    ;
                                08  DD 00013          PUSHL   #8                                         ;
           FEE5   CF            03  FB 00015          CALLS   #3, CHKPRO                                 ;
                  13            50  E8 0001A          BLBS    R0, 1$                                     ;
                  7E     0104   C2  3C 0001D          MOVZWL  260(R2), -(SP)                             ; 1360
                         00D0   C2  DD 00022          PUSHL   208(R2)                                    ;
                                04  DD 00026          PUSHL   #4                                         ;
           FED2   CF            03  FB 00028          CALLS   #3, CHKPRO                                 ;
                  04            50  E9 0002D          BLBC    R0, 2$                                     ;
                  50            01  D0 00030 1$:      MOVL    #1, R0                                     ; 1362
                                    04 00033          RET                                                ;
                                50  D4 00034 2$:      CLRL    R0                                         ; 1366
                                    04 00036          RET                                                ;

; Routine Size:  55 bytes,    Routine Base:  CODE + 0101
;  334          1367  1 END
;  335          1368  0 ELUDOM



;                                 PSECT SUMMARY
;
;       Name                       Bytes                      Attributes
;
;  COMMON                            5024  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  OVR,NOPIC,ALIGN(2)
;  CODE                               312  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                            Library Statistics
;
;                                              -------- Symbols --------      Pages      Processing
;       File                                   Total    Loaded  Percent       Mapped     Time
;
;  _$255$DUA28:[SYSLIB]LIB.L32;1              18619        43        0        1000        00:01.4



;                                  COMMAND QUALIFIERS

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

; Size:          312 code + 5024 data bytes
; Run Time:         00:13.7
; Elapsed Time:     00:50.7
; Lines/CPU Min:     5991
; Lexemes/CPU-Min: 49703
; Memory Used:  212 pages
; Compilation Complete
```

0191 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



BATCH
LIS

BROADCAST
LIS

BUFFERS
LIS

CONTROL
LIS

CHECKPROT
LIS

ASYNCHRON
LIS