F 1

```
DDDDDDDDDDDD     EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUU         UUU     GGGGGGGGGGGG
DDDDDDDDDDDD     EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUU         UUU     GGGGGGGGGGGG
DDDDDDDDDDDD     EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUU         UUU     GGGGGGGGGGGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG
DDD         DDD  EEEEEEEEEEEE     BBBBBBBBBBBB     UUU         UUU  GGG
DDD         DDD  EEEEEEEEEEEE     BBBBBBBBBBBB     UUU         UUU  GGG
DDD         DDD  EEEEEEEEEEEE     BBBBBBBBBBBB     UUU         UUU  GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG   GGGGGGGGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG   GGGGGGGGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG   GGGGGGGGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG         GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG         GGG
DDD         DDD  EEE              BBB         BBB  UUU         UUU  GGG         GGG
DDDDDDDDDDDD     EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUUUUUUUUUUUUUU     GGGGGGGGG
DDDDDDDDDDDD     EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUUUUUUUUUUUUUU     GGGGGGGGG
DDDDDDDDDDDD     EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUUUUUUUUUUUUUU     GGGGGGGGG
```

FILEID**SSIK   B 3   SS
VC

```
  SSSSSSSS   SSSSSSSS   IIIIII   KK      KK
  SSSSSSSS   SSSSSSSS   IIIIII   KK      KK
SS         SS             II     KK      KK
SS         SS             II     KK      KK
SS         SS             II     KK    KK
SS         SS             II     KK    KK
  SSSSSS     SSSSSS       II     KKKKKK
  SSSSSS     SSSSSS       II     KKKKKK
        SS         SS     II     KK    KK
        SS         SS     II     KK    KK
        SS         SS     II     KK      KK      ....
        SS         SS     II     KK      KK      ....
SSSSSSSS   SSSSSSSS     IIIIII   KK      KK      ....
SSSSSSSS   SSSSSSSS     IIIIII   KK      KK      ....

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
```

```
0001 0 MODULE SSIK (IDENT = 'V04-000') =
0002 1 BEGIN
0003 1
0004 1 !
0005 1 !****************************************************************************
0006 1 !*                                                                          *
0007 1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0008 1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0009 1 !*  ALL RIGHTS RESERVED.                                                    *
0010 1 !*                                                                          *
0011 1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0012 1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0013 1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0014 1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0015 1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0016 1 !*  TRANSFERRED.                                                            *
0017 1 !*                                                                          *
0018 1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0019 1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0020 1 !*  CORPORATION.                                                            *
0021 1 !*                                                                          *
0022 1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0023 1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0024 1 !*                                                                          *
0025 1 !*                                                                          *
0026 1 !****************************************************************************
0027 1
0028 1 !++
0029 1 ! FACILITY:  VAX/VMS System Service Call Monitor
0030 1 !
0031 1 ! ABSTRACT:
0032 1 !
0033 1 !       This module makes a copy of System Service vector in P0 space,
0034 1 !       then modifies the System Service vector JSB into intercept code.
0035 1 !
0036 1 !       SSI.B32 is split into 2 portions: SSIK.B32 is strictly running
0037 1 !       in kernel mode to do the setup, SSIU.B32 is running in user
0038 1 !       mode only.
0039 1 !
0040 1 ! ENVIRONMENT:
0041 1 !
0042 1 !       VAX/VMS operating system, CMKRNL privilege required.
0043 1 !
0044 1 ! AUTHOR:  David Thiel, 30-Dec-1981
0045 1 !
0046 1 ! Modified by:
0047 1 !
0048 1 !          Ping Sager,  19-Sep-1983
0049 1 !
0050 1 !--
0051 1
0052 1 !
0053 1 ! Include files
0054 1 !
0055 1 LIBRARY 'SYS$LIBRARY:LIB.L32';          ! VAX/VMS common definitions
0056 1 REQUIRE 'SRC$:SSIDEF.REQ';              ! Definitions for SSI
0146 1
```

```
!
! Table of contents
!
FORWARD ROUTINE
    ssik_start,                         ! Main routine
    make_p0_space,                      ! Allocate save/data/code area in P0
    ssik_setup;                         ! Establish intercept

GLOBAL
    ssv_munged_flag: INITIAL(0),        ! Set to TRUE when we actually
                                        !   change the system service vector.
    base : REF BBLOCK,                  ! Address of system service vector
    intercept : REF VECTOR [, BYTE],    ! Address of saved system vector,
                                        !   data, code area in P0
    range : VECTOR [2, LONG];           ! Pages allocated in P0
                                        !   (maps in ISSH.MAR)

! This portion (SSIK.B32) of the SSI.B32 is stictly running in kernel mode,
! and we are not intercepting anything in kernel mode.  So there is no
! need to have this flag to indicate this program is running.  This flag
! is set to indicate the other part (SSIU.B32) is running which runs in
! user mode.
!
EXTERNAL                                ! Flag set to indicate (SSIU.B32) is
    ssi_running_flag;                   ! running

OWN
                                        ! Variables (good for testing usage)
    l_base,                             ! (resident) copy of base
    l_intercept,                        ! (resident) copy of intercept
    l_tvl,                              ! (resident) copy of pointer
    range1 : VECTOR [2, LONG],          ! Maps in data area in range
    range2 : VECTOR [2, LONG];          ! Created virtual space over system
                                        !   service vector

!
! External routines
!
EXTERNAL ROUTINE
    SSI_USSK : ADDRESSING_MODE (GENERAL),
    sys$cretva : ADDRESSING_MODE (ABSOLUTE),    ! Create virtual address space
    sys$qiow : ADDRESSING_MODE (ABSOLUTE),      ! Base of transfer vector
    sys$rundwn : ADDRESSING_MODE (ABSOLUTE),    ! Rundown
    issh_entry,                                 ! Intercept code entry (ISSH)
    reset_ssv;                                  ! Clean up the mess

EXTERNAL LITERAL
    issh_vec_length;                    ! Length of monitor code (ISSH)

EXTERNAL
    ctl$gl_ctlbasva : ADDRESSING_MODE (ABSOLUTE),! Not used, if set, P0
                                        ! won't go away after image rundown
    issh_data_beg,                      ! Begin data area (template)
    issh_data_end,                      ! End data area (template)
    issh_prio_mask,                     ! Mask to control the calling of the
                                        !  user routines
    issh_vec_base,                      ! ISSH base address
```

```
;  116        0204  1      issh_running_flag,                  ! Contain pointer to ssi_running_flag
;  117        0205  1                                          !  in P0
;  118        0206  1      issh_stack,                         ! Local stack in P0 to keep track
;  119        0207  1                                          !  which user routine is active
;  120        0208  1                                          !  at the moment
;  121        0209  1      issh_stkptr,                        ! Pointer to the above local stack
;  122        0210  1      ssi_table : VECTOR [, LONG];        ! Configuration SS data table
;  123        0211  1
```

```
1 GLOBAL ROUTINE ssik_start (RUNDWN_ADDR) =
1
1 !---
1 !
1 ! Function:
1 !
1 !       This is the main routine of the VAX/VMS System Service
1 !       Monitor.  It calls appropriate actions.
1 !
1 ! Inputs:
1 !
1 !       RUNDWN_ADDR : This address only can be rundown system index.
1 !
1 ! Outputs:
1 !
1 !       Worst status encountered.
1 !
1 !---
1
2    BEGIN
2
2    BUILTIN FP;
2
2    LOCAL
2       prio_mask: REF VECTOR[,BYTE],          ! Current mask value after set
2       status;                                ! Return status
2
2
2    ! Check for rundown case.  The only way for this case to show up:
2    ! SYS$RUNDWN is called when image exits and intercept system service
2    ! is setup.  We simply put SSV back, and delete P0 space.
2    !
2    ! Note: next line is temporary, for I use the last 4 longwords in
2    ! SSV itself to store some values.
2    intercept = .(SYS$QIOW + sgn$c_sysvecpgs * 512 - 4);
2    IF .rundwn_addr EQL SYS$RUNDWN
2    THEN
3       BEGIN
3       IF .intercept EQL 0 THEN RETURN 0;     ! Can't be possible.
3       status = reset_ssv();                  ! Call routine to Clean up.
3       RETURN .status;                        ! Return Status. (Actually
2       END;                                   !  status is always 1).
2
2
2    ! If this code is first time called, sets up the intercept, else simply
2    ! returns.
2    !
2    IF .intercept NEQ 0 THEN RETURN ss$_normal;
2
2
2    ! P0 space has not been set up by anyone yet.  Grap some space.
2    ! Set up SSV.
2    !
2    base = sys$qiow;
2    status = make_p0_space();
2    IF .status THEN status = ssik_setup();
2    IF NOT .status THEN RETURN .status;
```

```
;  182       0269  2
;  183       0270  2
;  184       0271  2      ! Now that we have modified the system service vector, set the global flag
;  185       0272  2      ! which indicates that the system service vector has been modified. This
;  186       0273  2      ! flag gets cleared in RESETSSI.
;  187       0274  2      !
;  188       0275  2      ssv_munged_flag = 1;
;  189       0276  2
;  190       0277  2
;  191       0278  2      ! Initialize current mask to 0.  (Assume nothing is active at
;  192       0279  2      ! this moment.
;  193       0280  2      !
;  194       0281  2      prio_mask = .intercept + issh_prio_mask - issh_vec_base;
;  195       0282  2      .prio_mask = 0;
;  196       0283  2      RETURN ss$_normal;
;  197       0284  1      END;


                                                           .TITLE  SSIK
                                                           .IDENT  \V04-000\

                                                           .PSECT  $OWN$,NOEXE,2

                                       00000 L_BASE: .BLKB    4
                                       00004 L_INTERCEPT:
                                                     .BLKB    4
                                       00008 L_TVL:  .BLKB    4
                                       0000C RANGE1: .BLKB    8
                                       00014 RANGE2: .BLKB    8

                                                           .PSECT  $GLOBAL$,NOEXE,2

                             00000000  00000 SSV_MUNGED_FLAG::
                                                     .LONG    0                                ;
                                       00004 BASE::  .BLKB    4
                                       00008 INTERCEPT::
                                                     .BLKB    4
                                       0000C RANGE:: .BLKB    8

                                                           .EXTRN  SSI_RUNNING_FLAG
                                                           .EXTRN  SSI_USSK, SYS$CRETVA
                                                           .EXTRN  SYS$QIOW, SYS$RUNDWN
                                                           .EXTRN  ISSH_ENTRY, RESET_SSV
                                                           .EXTRN  ISSH_VEC_LENGTH
                                                           .EXTRN  CTL$GL_CTLBASVA
                                                           .EXTRN  ISSH_DATA_BEG, ISSH_DATA_END
                                                           .EXTRN  ISSH_PRIO_MASK, ISSH_VEC_BASE
                                                           .EXTRN  ISSH_RUNNING_FLAG
                                                           .EXTRN  ISSH_STACK, ISSH_STKPTR
                                                           .EXTRN  SSI_TABLE

                                                           .PSECT  $CODE$,NOWRT,2

                                  0004 00000               .ENTRY  SSIK_START, Save R2                      ; 0212
                   52     0000'   CF  9E 00002             MOVAB   INTERCEPT, R2                            ;
                   62 00000000G   9F  D0 00007             MOVL    @#SYS$QIOW+2556, INTERCEPT               ; 0246
        00000000G  8F        04   AC  D1 0000E             CMPL    RUNDWN_ADDR, #SYS$RUNDWN                 ; 0247
```

```
                                0A  12 00016          BNEQ    1$                                    ;
                                62  D5 00018          TSTL    INTERCEPT                             ; 0250
                                3C  13 0001A          BEQL    3$                                    ;
           0000G  CF            00  FB 0001C          CALLS   #0, RESET_SSV                         ; 0251
                                    04 00021          RET                                           ; 0252
                                62  D5 00022  1$:     TSTL    INTERCEPT                             ; 0259
                                2E  12 00024          BNEQ    2$                                    ;
             FC   A2 00000000G  8F  D0 00026          MOVL    #SYS$QIOW, BASE                       ; 0265
           0000V  CF            00  FB 0002E          CALLS   #0, MAKE_P0_SPACE                     ; 0266
                  24            50  E9 00033          BLBC    STATUS, 4$                            ; 0267
           0000V  CF            00  FB 00036          CALLS   #0, SSIK_SETUP                        ;
                  1C            50  E9 0003B          BLBC    STATUS, 4$                            ; 0268
             F8   A2            01  D0 0003E          MOVL    #1, SSV_MUNGED_FLAG                   ; 0275
                  50     0000G  CF  9E 00042          MOVAB   ISSH_PRIO_MASK, R0                    ; 0281
                  50            62  C0 00047          ADDL2   INTERCEPT, R0                         ;
                  51     0000G  CF  9E 0004A          MOVAB   ISSH_VEC_BASE, R1                     ;
                  50            51  C2 0004F          SUBL2   R1, PRIO_MASK                         ;
                                60  D4 00052          CLRL    (PRIO_MASK)                           ; 0282
                  50            01  D0 00054  2$:     MOVL    #1, R0                                ; 0283
                                    04 00057          RET                                           ;
                                50  D4 00058  3$:     CLRL    R0                                    ; 0284
                                    04 0005A  4$:     RET                                           ;

; Routine Size:  91 bytes,    Routine Base:  $CODE$ + 0000
```

```
  1 ROUTINE make_p0_space =
  1
  1 !---
  1 !
  1 ! Function:
  1 !
  1 !       Create a save area for intercepting system services in P0
  1 !       space.
  1 !
  1 ! Inputs:
  1 !
  1 !       None.
  1 !
  1 ! Outputs:
  1 !
  1 !       status is returned.
  1 !
  1 !---
  1
  2     BEGIN
  2
  2     BIND
  2         exp_size = (issh_vec_length+%X'1FF') ^ -9;
  2                                           ! ISSH.MAR code side
  2
  2     LOCAL
  2         status;                           ! Return status
  2
  2
  2     ! Create a save area to save the system vector, data area, and code
  2     ! in P0 space.
  2     !
P 2     status = $EXPREG (
P 2         PAGCNT = exp_size,                ! pages to create
P 2         REGION = 0,                       ! P0 region
P 2         ACMODE = psl$c_kernel,            ! kernel mode to own pages
  2         RETADR = range);                  ! range of allocated addresses
  2     IF NOT .status THEN RETURN .status;
  2
  2
  2     ! Map in from ISSH.MAR.
  2     !
  2     CH$MOVE (issh_vec_length, issh_vec_base, .range[0]);
  2
  2
  2     ! Set protection to saved area.
  2     !
P 2     status = $SETPRT (
P 2         INADR = range,                    ! pages to protect
  2         PROT = prt$c_urkw);               ! kernal writable, others can read
  2     IF NOT .status THEN RETURN .status;
  2
  2
  2     ! Mapped in data area and control area.
  2     !
  2     range1 [0] = .range [0] + issh_data_beg - issh_vec_base;
  2     range1 [1] = .range [0] + issh_data_end - issh_vec_base - 1;
```

```
                0342  2
                0343  2
                0344  2        ! Set protection to data area.
                0345  2        !
              P 0346  2        status = $SETPRT (
              P 0347  2            INADR = range1,                  ! pages to protect
                0348  2            PROT = prt$c_uw);                ! everybody can access data pages
                0349  2        IF NOT .status THEN RETURN .status;
                0350  2
                0351  2
                0352  2        ! Set the pointer to saved area in P0 space.
                0353  2        !
                0354  2        intercept = .range [0];
                0355  2        RETURN ss$_normal;
                0356  2
                0357  1        END;



                                                      .EXTRN  SYS$EXPREG, SYS$SETPRT

                                  03FC 00000 MAKE_P0_SPACE:
                                                      .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9                       ; 0285
                      59       0000G   CF  9E 00002   MOVAB   ISSH_VEC_BASE, R9                                  ;
                      58   00000000G   00  9E 00007   MOVAB   SYS$SETPRT, R8                                     ;
                      57       0000'   CF  9E 0000E   MOVAB   RANGE, R7                                          ;
                                       7E  7C 00013   CLRQ    -(SP)                                              ; 0321
                                       57  DD 00015   PUSHL   R7                                                 ;
                           00000000*   8F  DD 00017   PUSHL   #<<ISSH_VEC_LENGTH+511>@-9>                        ;
             00000000G    00           04  FB 0001D   CALLS   #4, SYS$EXPREG                                     ;
                          56           50  D0 00024   MOVL    R0, STATUS                                         ;
                          4E           56  E9 00027   BLBC    STATUS, 1$                                         ; 0322
     00    B7             69   0000G   8F  28 0002A   MOVC3   #ISSH_VEC_LENGTH, ISSH_VEC_BASE, @RANGE            ; 0327
                          7E           0E  7D 00031   MOVQ    #14, -(SP)                                         ; 0334
                                       7E  7C 00034   CLRQ    -(SP)                                              ;
                                       57  DD 00036   PUSHL   R7                                                 ;
                          68           05  FB 00038   CALLS   #5, SYS$SETPRT                                     ;
                          56           50  D0 0003B   MOVL    R0, STATUS                                         ;
                          37           56  E9 0003E   BLBC    STATUS, 1$                                         ; 0335
                          50   0000G   CF  9E 00041   MOVAB   ISSH_DATA_BEG, R0                                  ; 0340
                          50           67  C0 00046   ADDL2   RANGE, R0                                          ;
                          51           69  9E 00049   MOVAB   ISSH_VEC_BASE, R1                                  ;
   0000'   CF             50           51  C3 0004C   SUBL3   R1, R0, RANGE1                                     ;
                          50   0000G   CF  9E 00052   MOVAB   ISSH_DATA_END, R0                                  ; 0341
                          50           67  C0 00057   ADDL2   RANGE, R0                                          ;
                          51           69  9E 0005A   MOVAB   ISSH_VEC_BASE, R1                                  ;
                          50           51  C2 0005D   SUBL2   R1, R0                                             ;
                0000'     CF      FF   A0  9E 00060   MOVAB   -1(R0), RANGE1+4                                   ;
                          7E           04  7D 00066   MOVQ    #4, -(SP)                                          ; 0348
                                       7E  7C 00069   CLRQ    -(SP)                                              ;
                               0000'   CF  9F 0006B   PUSHAB  RANGE1                                             ;
                          68           05  FB 0006F   CALLS   #5, SYS$SETPRT                                     ;
                          56           50  D0 00072   MOVL    R0, STATUS                                         ;
                          04           56  E8 00075   BLBS    STATUS, 2$                                         ; 0349
                          50           56  D0 00078 1$:  MOVL  STATUS, R0                                        ;
                                           04 0007B   RET                                                        ;
                  FC      A7           67  D0 0007C 2$:  MOVL  RANGE, INTERCEPT                                  ; 0354
```

```
                         50            01  D0 00080          MOVL    #1, R0                                  ; 0355
                                           04 00083          RET                                             ; 0357

; Routine Size:  132 bytes,    Routine Base:  $CODE$ + 005B

;  272          0358  1
```

```
1 ROUTINE ssik_setup : PSECT (lkcode_1) =
1
1 !---
1 !
1 ! Function:
1 !
1 !       Setup System Service Intercept.  Vector to be intercepted is
1 !       in base, save/data area is in intercept.
1 !
1 ! Inputs:
1 !
1 !       Entry point address of this routine.
1 !
1 ! Outputs:
1 !
1 !       status is returned.
1 !
1 !---
1
2     BEGIN
2
2     LOCAL
2         old_stat,                       ! Old AST enable status
2         status,                         ! Return status
2         temp_vec: VECTOR[2];            ! Parameter for $SETPRT
2
2     BIND
2         tvl = base [sgn$c_sysvecpgs * 512, 0, 0, 0] : BBLOCK FIELD (tvb);
2
2
2     l_base = .base;                     ! SSV base address
2     l_intercept = .intercept;           ! Copied SSV base address
2     l_tvl = tvl;                        ! End of SSV
2
2
2     ! Save original system vector in saved area.  Disable AST first.
2     !
2     return_if_error (old_stat = $SETAST (ENBFLG = 0));
2     CH$MOVE (sgn$c_sysvecpgs*512, .l_base, .l_intercept);
2
2
2     ! Create virtual memory over the original system vector and copy
2     ! the original contents back into it.
2     !
2     range2 [0] = .l_base;
2     range2 [1] = .l_base + sgn$c_sysvecpgs*512 - 1;
2     status = (sys$cretva + (%X'80000000' - sys$qiow)) (
2         range2,                 ! inadr
2         range2,                 ! retadr
2         0);                     ! acmode?
2
2     IF NOT .status
2     THEN
3         BEGIN
3
3
3         ! Enable AST.
```

```
  3             !
  3             IF .old_stat EQL ss$_wasset
  3             THEN
  3                 return_if_error ($SETAST (ENBFLG = 1));
  3
  3             RETURN .status;
  2             END;
  2
  2
  2         ! restore original contents of save/data area
  2         !
  2         CH$MOVE (sgn$c_sysvecpgs*512, .l_intercept, .l_base);
  2
  2
  2         ! Set protection.
  2         !
P 2         status = $SETPRT (
P 2             INADR = range2,                 ! pages to protect
  2             PROT = prt$c_urkw);             ! kernal writable, others can read
  2
  2         IF NOT .status
  2         THEN
  3             BEGIN
  3
  3
  3             ! Enable AST.
  3             !
  3             IF .old_stat EQL ss$_wasset
  3             THEN
  3                 return_if_error ($SETAST (ENBFLG = 1));
  3
  3             RETURN .status;
  2             END;
  2
  2
  2         ! Initialize local stack in P0.
  2         ! Stack pointer is 1 (points to the 1st element on stack),
  2         ! stack value is 0 (nothing is active at the moment).
  2         !
  2         .l_intercept + issh_stkptr - issh_vec_base = 1;
  2         .l_intercept + issh_stack - issh_vec_base = 0;
  2
  2
  2         ! Make the running flag user-writable.
  2         !
  2         temp_vec[0] = ssi_running_flag;
  2         temp_vec[1] = ssi_running_flag;
  2         return_if_error ($SETPRT (INADR = temp_vec, PROT = prt$c_uw));
  2
  2
  2         ! Set up the running flag.  If this flag is set, means that SSIU.B32
  2         ! is running so won't intercept any system service from that program.
  2         ! Otherwise, go ahead to intercept.  For SSIU.B32 is running in user
  2         ! mode, which is the only mode we intercept.
  2         !
  2         .l_intercept + issh_running_flag - issh_vec_base = ssi_running_flag;
  2
```

```
;  388          0473  2
;  389          0474  2      ! Set up 3 pointers at the end of the System Service Vector.
;  390          0475  2      !   - A pointer to the saved system service vector in P0.
;  391          0476  2      !   - A pointer to the saved intercept code entry point.
;  392          0477  2      !   - Address of the user defined system service.
;  393          0478  2      !
;  394          0479  2      tvl [ptr] = .l_intercept;
;  395          0480  2      tvl [pg0] = .l_intercept + issh_entry - issh_vec_base;
;  396          0481  2      tvl [pg1] = SST_USSK;
;  397          0482  2
;  398          0483  2      DECR i FROM (.ssi_table[-1])/2 -1 TO 0 DO
;  399          0484  3          BEGIN
;  400          0485  3          BIND
;  401          0486  3              p = ssi_table [.i*2] : VECTOR [, LONG],
;  402          0487  3              t = .l_base + .p [1] : BBLOCK;
;  403          0488  3
;  404          0489  3
;  405          0490  3          ! Verify it is in System space.
;  406          0491  3          !
;  407          0492  3          IF .p [1] GEQU %X'800'
;  408          0493  3          THEN
;  409          0494  3              0
;  410          0495  3
;  411          0496  3          ELSE
;  412          0497  3
;  413          0498  3
;  414          0499  3              ! CALLS/CALLG intercepted at entry
;  415          0500  3              !
;  416          0501  4              BEGIN
;  417          0502  4              t [2, 0, 8, 0] = op$_jsb;    ! JSB
;  418          0503  4              t [3, 0, 8, 0] = %X'DF';     ! @W^ addressing mode
;  419          0504  4              t [4, 0, 16, 0] = tvl [pg0] - t [6, 0, 8, 0];
;  420          0505  3              END;
;  421          0506  2          END;                             ! End of DECR.
;  422          0507  2
;  423          0508  2
;  424          0509  2      ! Enable AST.
;  425          0510  2      !
;  426          0511  2      IF .old_stat EQL ss$_wasset
;  427          0512  2      THEN
;  428          0513  2          return_if_error ($SETAST (ENBFLG = 1));
;  429          0514  2
;  430          0515  2      RETURN ss$_normal;
;  431          0516  2
;  432          0517  1      END;


                                                                   .EXTRN  SYS$SETAST

                                                                   .PSECT  LKCODE_1,NOWRT,  SHR,2

                                               0FFC 00000 SSIK_SETUP:
                                                                   .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11          ; 0359
                                5B 00000000G  00  9E 00002         MOVAB   SYS$SETPRT, R11                               ;
                                5A 00000000G  00  9F 00009         MOVAB   SYS$SETAST, R10                               ;
                                59     0000'  CF  9E 00010         MOVAB   L_INTERCEPT, R9                               ;
```

```
                              5E                 08  C2 00015         SUBL2    #8, SP                                          ;
            56       0000'    CF  00000A00       8F  C1 00018         ADDL3    #2560, BASE, R6                                 ; 0386
                       FC     A9       0000'     CF  7D 00022         MOVQ     BASE, L_BASE                                    ; 0389
                       04     A9                 56  D0 00028         MOVL     R6, L_TVL                                       ; 0391
                                                 7E  D4 0002C         CLRL     -(SP)                                           ; 0396
                              6A                 01  FB 0002E         CALLS    #1, SYS$SETAST                                  ;
                              58                 50  D0 00031         MOVL     R0, OLD_STAT                                    ;
                              5B                 50  E9 00034         BLBC     STATUS, 2$                                      ;
                              57          FC     A9  D0 00037         MOVL     L_BASE, R7                                      ; 0397
      00    B9                67        0A00     8F  28 0003B         MOVC3    #2560, (R7), @L_INTERCEPT                       ;
                       10     A9                 57  D0 00042         MOVL     R7, RANGE2                                      ; 0403
                       14     A9        09FF     C7  9E 00046         MOVAB    2559(R7), RANGE2+4                              ; 0404
                                                 7E  D4 0004C         CLRL     -(SP)                                           ; 0405
                                          10     A9  9F 0004E         PUSHAB   RANGE2                                          ;
                                          10     A9  9F 00051         PUSHAB   RANGE2                                          ;
                  00000000*   9F                 03  FB 00054         CALLS    #3, @#<SYS$CRETVA+<-2147483648-SYS$QIOW>>       ;
                              57                 50  D0 0005B         MOVL     R0, STATUS                                      ;
                              0E                 57  E8 0005E         BLBS     STATUS, 1$                                      ; 0410
                              09                 58  D1 00061         CMPL     OLD_STAT, #9                                    ; 0417
                                                 2F  12 00064         BNEQ     3$                                              ;
                                                 01  DD 00066         PUSHL    #1                                              ; 0419
                              6A                 01  FB 00068         CALLS    #1, SYS$SETAST                                  ;
                              27                 50  E8 0006B         BLBS     STATUS, 3$                                      ;
                                                     04 0006E         RET                                                      ; 0421
      FC    B9         00     B9        0A00     8F  28 0006F  1$:    MOVC3    #2560, @L_INTERCEPT, @L_BASE                    ; 0427
                              7E                 0E  7D 00077         MOVQ     #14, -(SP)                                      ; 0434
                                                 7E  7C 0007A         CLRQ     -(SP)                                           ;
                                          10     A9  9F 0007C         PUSHAB   RANGE2                                          ;
                              6B                 05  FB 0007F         CALLS    #5, SYS$SETPRT                                  ;
                              57                 50  D0 00082         MOVL     R0, STATUS                                      ;
                              11                 57  E8 00085         BLBS     STATUS, 4$                                      ; 0436
                              09                 58  D1 00088         CMPL     OLD_STAT, #9                                    ; 0443
                                                 08  12 0008B         BNEQ     3$                                              ;
                                                 01  DD 0008D         PUSHL    #1                                              ; 0445
                              6A                 01  FB 0008F         CALLS    #1, SYS$SETAST                                  ;
                              3F                 50  E9 00092  2$:    BLBC     STATUS, 5$                                      ;
                              50                 57  D0 00095  3$:    MOVL     STATUS, R0                                      ; 0447
                                                     04 00098         RET                                                      ;
                              50       0000G     CF  9E 00099  4$:    MOVAB    ISSH_STKPTR, R0                                 ; 0455
                              50                 69  C0 0009E         ADDL2    L_INTERCEPT, R0                                 ;
                              51       0000G     CF  9E 000A1         MOVAB    ISSH_VEC_BASE, R1                               ;
                              50                 51  C2 000A6         SUBL2    R1, R0                                          ;
                              60                 01  D0 000A9         MOVL     #1, (R0)                                        ;
                              50       0000G     CF  9E 000AC         MOVAB    ISSH_STACK, R0                                  ; 0456
                              50                 69  C0 000B1         ADDL2    L_INTERCEPT, R0                                 ;
                              51       0000G     CF  9E 000B4         MOVAB    ISSH_VEC_BASE, R1                               ;
                              50                 51  C2 000B9         SUBL2    R1, R0                                          ;
                                                 60  D4 000BC         CLRL     (R0)                                            ;
                              6E       0000G     CF  9E 000BE         MOVAB    SSI_RUNNING_FLAG, TEMP_VEC                      ; 0461
                       04     AE       0000G     CF  9E 000C3         MOVAB    SSI_RUNNING_FLAG, TEMP_VEC+4                    ; 0462
                              7E                 04  7D 000C9         MOVQ     #4, -(SP)                                       ; 0463
                                                 7E  7C 000CC         CLRQ     -(SP)                                           ;
                                          10     AE  9F 000CE         PUSHAB   TEMP_VEC                                        ;
                              6B                 05  FB 000D1         CALLS    #5, SYS$SETPRT                                  ;
                              7C                 50  E9 000D4  5$:    BLBC     STATUS, 9$                                      ;
                              50                 69  D0 000D7         MOVL     L_INTERCEPT, R0                                 ; 0471
                              51       0000GCF40     9E 000DA         MOVAB    ISSH_RUNNING_FLAG[R0], R1                       ;
```

```
                                     52       0000G  CF   9E 000E0          MOVAB   ISSH_VEC_BASE, R2                      ;
                                     51              52   C2 000E5          SUBL2   R2, R1                                 ;
                                     61       0000G  CF   9E 000E8          MOVAB   SSI_RUNNING_FLAG, (R1)                 ;
                              FC     A6              50   D0 000ED          MOVL    R0, -4(R6)                             ; 0479
                                     51       0000GCF40   9E 000F1          MOVAB   ISSH_ENTRY[R0], R1                     ; 0480
                                     50       0000G  CF   9E 000F7          MOVAB   ISSH_VEC_BASE, R0                      ;
                 F8     A6           51              50   C3 000FC          SUBL3   R0, R1, -8(R6)                         ;
                              F4     A6 000000000G   00   9E 00101          MOVAB   SSI_USSK, -'2(R6)                      ; 0481
                        51      0000G  CF            02   C7 00109          DIVL3   #2, SSI_TABLE-4, R1                    ; 0483
                                     50              51   D0 0010F          MOVL    R1, I                                  ; 0487
                                                     2C   11 00112          BRB     7$                                     ;
                        51           50              01   78 00114 6$:      ASHL    #1, I, R1                              ; 0486
                                     52       0000GCF41   DE 00118          MOVAL   SSI_TABLE[R1], R2                      ;
                        51           FC      A9      04   A2 C1 0011E       ADDL3   4(R2), L_BASE, R1                      ; 0487
                             00000800        8F      04   A2 D1 00124       CMPL    4(R2), #2048                           ; 0492
                                                     12   1E 0012C          BGEQU   7$                                     ;
                              02     A1       DF16   8F   B0 0012E          MOVW    #57110, 2(R1)                          ; 0502
                                     52         F8   A6   9E 00134          MOVAB   -8(R6), R2                             ; 0504
                                     52              51   C2 00138          SUBL2   R1, R2                                 ;
                 04     A1           52              06   A3 0013B          SUBW3   #6, R2, 4(R1)                          ;
                                     D1              50   F4 00140 7$:      SOBGEQ  I, 6$                                  ; 0483
                                     09              58   D1 00143          CMPL    OLD_STAT, #9                           ; 0511
                                                     08   12 00146          BNEQ    8$                                     ;
                                                     01   DD 00148          PUSHL   #1                                     ; 0513
                                     6A              01   FB 0014A          CALLS   #1, SYS$SETAST                         ;
                                     03              50   E9 0014D          BLBC    STATUS, 9$                             ;
                                     50              01   D0 00150 8$:      MOVL    #1, R0                                 ; 0515
                                                          04 00153 9$:      RET                                            ; 0517

; Routine Size:  340 bytes,    Routine Base:  LKCODE_1 + 0000


;  433          0518  1
;  434          0519  1 END
;  435          0520  0 ELUDOM



;                                  PSECT SUMMARY
;
;     Name                         Bytes                      Attributes
;
;  $GLOBAL$                           20  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $OWN$                              28  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                            223  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  LKCODE_1                          340  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                            Library Statistics
;
;                                         -------- Symbols --------       Pages       Processing
;     File                                Total    Loaded   Percent       Mapped      Time
```

```
;
;  _$255$DUA28:[SYSLIB]LIB.L32;1           18619       12        0      1000        00:01.8



;                                   COMMAND QUALIFIERS

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

; Size:          563 code + 48 data bytes
; Run Time:         00:12.6
; Elapsed Time:     00:40.1
; Lines/CPU Min:     2474
; Lexemes/CPU-Min: 15925
; Memory Used:  131 pages
; Compilation Complete
```

0101 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

SSIK
LIS

DELTA
LIS

DELETE

DELEMAIN
LIS

SSITAB
LIS

DELETE
MAP

DELTA

DELTA
MAP

SSIU
LIS

SSIUW
LIS

PURGE
LIS

STRUCDEF
LIS

SSIDISP
LIS

SØDELTA
MAP

DELETE
REQ