```
UUU          UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSS  YYY          YYY
UUU          UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSS  YYY          YYY
UUU          UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSS  YYY          YYY
UUU          UUU  EEE                    TTT        PPP          PPP  SSS              YYY          YYY
UUU          UUU  EEE                    TTT        PPP          PPP  SSS              YYY          YYY
UUU          UUU  EEE                    TTT        PPP          PPP  SSS              YYY          YYY
UUU          UUU  EEE                    TTT        PPP          PPP  SSS                 YYY    YYY
UUU          UUU  EEE                    TTT        PPP          PPP  SSS                 YYY    YYY
UUU          UUU  EEE                    TTT        PPP          PPP  SSS                 YYY    YYY
UUU          UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS              YYY
UUU          UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS              YYY
UUU          UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS              YYY
UUU          UUU  EEE                    TTT        PPP                        SSS              YYY
UUU          UUU  EEE                    TTT        PPP                        SSS              YYY
UUU          UUU  EEE                    TTT        PPP                        SSS              YYY
UUU          UUU  EEE                    TTT        PPP                        SSS              YYY
UUU          UUU  EEE                    TTT        PPP                        SSS              YYY
UUUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP                SSSSSSSSSSS              YYY
UUUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP                SSSSSSSSSSS              YYY
UUUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP                SSSSSSSSSSS              YYY
```

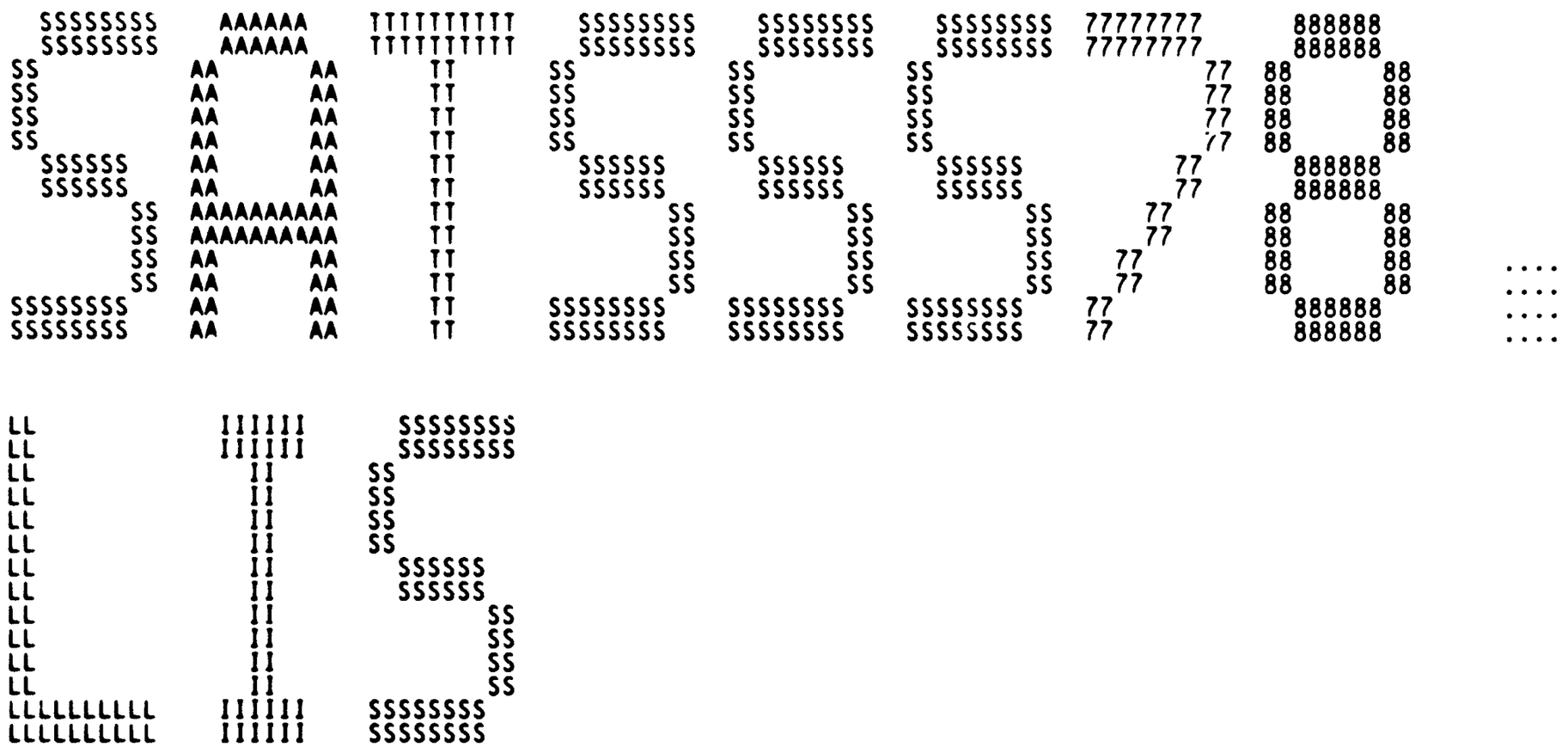
SSSSSSSS AAAAAA TTTTTTTTTT SSSSSSSS SSSSSSSS SSSSSSSS 77777777 888888
SSSSSSSS AAAAAA TTTTTTTTTT SSSSSSSS SSSSSSSS SSSSSSSS 77777777 888888
SS AA AA TT SS SS SS 77 88 88
SS AA AA TT SS SS SS 77 88 88
SS AA AA TT SS SS SS 77 88 88
SS AA AA TT SS SS SS 77 88 88
SSSSSS AA AA TT SSSSSS SSSSSS SSSSSS 77 888888
SSSSSS AA AA TT SSSSSS SSSSSS SSSSSS 77 888888
SS AAAAAAAAAA TT SS SS SS 77 88 88
SS AAAAAAAAAA TT SS SS SS 77 88 88
SS AA AA TT SS SS SS 77 88 88
SS AA AA TT SS SS SS 77 88 88
SSSSSSSS AA AA TT SSSSSSSS SSSSSSSS SSSSSSSS 77 888888
SSSSSSSS AA AA TT SSSSSSSS SSSSSSSS SSSSSSSS 77 888888
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

Table of contents

```
0000     1          .TITLE  SATSSS78 SATS SYSTEM SERV TESTS  $LK/ULWSET (SUCC S.C.)
0000     2          .IDENT  'V04-000'
0000     3
0000     4  ;
0000     5  ;*****************************************************************************
0000     6  ;*                                                                           *
0000     7  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     8  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9  ;*  ALL RIGHTS RESERVED.                                                     *
0000    10  ;*                                                                           *
0000    11  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
0000    12  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
0000    13  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
0000    14  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
0000    15  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
0000    16  ;*  TRANSFERRED.                                                             *
0000    17  ;*                                                                           *
0000    18  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
0000    19  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0000    20  ;*  CORPORATION.                                                             *
0000    21  ;*                                                                           *
0000    22  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
0000    23  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24  ;*                                                                           *
0000    25  ;*                                                                           *
0000    26  ;*****************************************************************************
0000    27  ;
0000    28
0000    29  ;++
0000    30  ; FACILITY:     SYSTST (SATS SYSTEM SERVICE TESTS)
0000    31  ;
0000    32  ; ABSTRACT:
0000    33  ;
0000    34  ;               THIS MODULE CONTAINS SUBROUTINES WHICH, WHEN LINKED
0000    35  ; WITH SUCCOMMON.OBJ, FORM TEST MODULE SATSSS78 TO TEST SUCCESSFUL
0000    36  ; OPERATION OF THE $LK/ULWSET SYSTEM SERVICES. EACH SERVICE IS INVOKED
0000    37  ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38  ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39  ; OPERATION OF EACH SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40  ; CHECKING FOR AN SS$_NORMAL STATUS CODE, EXPECTED RETURN ARGUMENTS
0000    41  ; AND EXPECTED FUNCTIONALITY PERFORMED.
0000    42  ;
0000    43  ; ENVIRONMENT:  USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000    44  ;               DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000    45  ;
0000    46  ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: APR, 1978
0000    47  ;
0000    48  ; MODIFIED BY:
0000    49  ;
0000    50  ;               . : VERSION
0000    51  ; 01     -
0000    52  ;--
```

```
                                J 3
0000     54          .SBTTL  DECLARATIONS
0000     55 ;
0000     56 ; INCLUDE FILES:
0000     57 ;
0000     58          $PRVDEF                         ; PRIVILEGE BIT DEFINITIONS
0000     59          $PHDDEF                         ; PROCESS HEADER OFFSETS
0000     60          $PSLDEF                         ; PROCESSOR STATUS LONGWORD DEFINITIONS
0000     61 ;
0000     62 ; MACROS:
0000     63 ;
0000     64 ;
0000     65 ; EQUATED SYMBOLS:
0000     66 ;
0000     67 ;
0000     68 ; OWN STORAGE:
0000     69 ;
```

```
00000000     7;          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
     0000    72 TEST_MOD_NAME:: STRING  C,<SATSSS78>     ; TEST MODULE NAME
     0009    73 TEST_MOD_NAME_D: STRING I,<SATSSS78>     ; TEST MODULE NAME DESCRIPTOR
     0019    74 MSG1_INP_CTL:   STRING  I,<  SSLUW!4ZW: CONDITIONS:>
     0039    75                                          ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
     0039    76 MSG3_ERR_CTL::  STRING  I,< *SSLUW!4ZW: !AS>
     0051    77                                          ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
```

```
                00000000     79          .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008  0000  00000008     80 PRIVMASK:        .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000C  0008  0000000C     81 PAGCNT_ERG:      .BLKL   1               ; PAGCNT ARGUMENT FOR EXPREG
00000010  000C  00000010     82 REGION_ERG:      .BLKL   1               ; REGION ARGUMENT FOR EXPREG
00000018  0010  00000018     83 RETADR:          .BLKQ   1               ; RETADR ARG FOR SUBJ $LKWSET & $ULWSET
```

```
                                        0018      .SBTTL  CONDITION TABLES
                                        0018  ;
                                        0018  ;       ***** CONDITION TABLES FOR LKWSET/ULWSET SYSTEM SERVICES *****
                                        0018  ;
                                        0018      COND    1,QUAD,<INADR>,-
                                        0018              <CURRENT IMAGE PAGE>,-
                                        0018              <PROGRAM REGION -- SINGLE PAGE>,-
                                        0018              <PROGRAM REGION -- MULTIPLE PAGES> -
                                        0018              <CONTROL REGION -- SINGLE PAGE>,-
                                        0018              <CONTROL REGION -- MULTIPLE PAGES>,-
                                        0018
                   00000000'00000000'  00C4                .LONG      TM_SETUP,TM_SETUP
 00000000'00000000'00000000'00000000'  00CC                .LONG      0[8]
 00000000'00000000'00000000'00000000'  00DC
                                        00EC  ;
                                        00EC      COND    2,NOTARG,<PAGE OWNER ACCESS MODE>,-
                                        00EC              <KERNEL>,-
                                        00EC              <EXECUTIVE>,-
                                        00EC              <SUPERVISOR>,-
                                        00EC              <USER>,-
                                        00EC
                             00000000  0135                .LONG      PSL$C_KERNEL
                             00000001  0139                .LONG      PSL$C_EXEC
                             00000002  013D                .LONG      PSL$C_SUPER
                             00000003  0141                .LONG      PSL$C_USER
                                        0145  ;
                                        0145      COND    3,LONG,<ACMODE>,-
                                        0145              <KERNEL>,-
                                        0145              <EXECUTIVE>,-
                                        0145              <SUPERVISOR>,-
                                        0145              <USER>,-
                                        0145
                             00000000  017E                .LONG      PSL$C_KERNEL
                             00000001  0182                .LONG      PSL$C_EXEC
                             00000002  0186                .LONG      PSL$C_SUPER
                             00000003  018A                .LONG      PSL$C_USER
                                        018E  ;
                                        018E      COND    4,NULL
                                        018F
                                        018F      COND    5,NULL
                                        0190
                                    00000000      .PSECT  SATSSS78,RD,WRT,EXE
```

```
                                        .SBTTL  TM_SETUP, TM_CLEANUP
                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;
                                ;               TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                                ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                                ; TEST MODULE EXECUTION.
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       BSBW TM_SETUP   BSBW TM_CLEANUP
                                ;
                                ; INPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT INPUTS:
                                ;
                                ;       NONE
                                ;
                                ; OUTPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT OUTPUTS:
                                ;
                                ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                                ;                  ALL PRIVILEGES ACQUIRED.
                                ;
                                ; COMPLETION CODES:
                                ;
                                ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                                ;
                                ; SIDE EFFECTS:
                                ;
                                ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                                ;       (VIA RSB) IF ERROR ENCOUNTERED.
                                ;
                                ;--
                                
                                
                                TM_SETUP::
                        52  D4 0000        CLRL    R2                      ; INITIALIZE
                        53  D4 0002        CLRL    R3                      ; .. CONDITION
                        54  D4 0004        CLRL    R4                      ; .... TABLE
                        55  D4 0006        CLRL    R5                      ; ...... INDEX
                        56  D4 0008        CLRL    R6                      ; ........ REGISTERS
                    FFF3'   30 000A        BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE BEGIN MSG
00000000'EF     00000000'EF   DE 000D        MOVAL   TEST_MOD_SUCC,TMD_ADDR  ; ASSUME END MSG WILL SHOW SUCCESS
    03  00  00000000'8F   F0 0018        INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
            00000000'EF       0020
                              0025        MODE    TO,S$,KRNL              ; KERNEL MODE TO ACCESS PHD
        59  00000000'9F   D0 0048        MOVL    @#CTL$GL_PHD,R9         ; GET PROCESS HEADER ADDRESS
        00000000'EF   69  DE 004F        MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                              0056        MODE    FROM,S$ ; BACK TO USER MODE
                              0057        PRIV    ADD,ALL                 ; GET ALL PRIVILEGES
```

```
                                                B 4
                     0077   183          $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                     0084   184          SS_CHECK  NORMAL                ; CHECK STATUS CODE RETURNED FROM SETPRN
                  05 00B2   185          RSB                             ; RETURN TO MAIN ROUTINE
                     00B3   186 TM_CLEANUP::
              FF4A' 30 00B3   187          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE END MSG
                  05 00B6   188          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                     00B7   190          .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
                     00B7   191 ;++
                     00B7   192 ; FUNCTIONAL DESCRIPTION:
                     00B7   193 ;
                     00B7   194 ;                 CONDX AND CONDX_CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
                     00B7   195 ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
                     00B7   196 ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
                     00B7   197 ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
                     00B7   198 ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
                     00B7   199 ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
                     00B7   200 ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
                     00B7   201 ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
                     00B7   202 ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
                     00B7   203 ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
                     00B7   204 ;
                     00B7   205 ; CALLING SEQUENCE:
                     00B7   206 ;
                     00B7   207 ;       BSBW CONDX   BSBW CONDX_CLEANUP
                     00B7   208 ;         WHERE X = 1,2,3,4,5
                     00B7   209 ;
                     00B7   210 ; INPUT PARAMETERS:
                     00B7   211 ;
                     00B7   212 ;       CONFLICT = 0
                     00B7   213 ;
                     00B7   214 ; IMPLICIT INPUTS:
                     00B7   215 ;
                     00B7   216 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                     00B7   217 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                     00B7   218 ;
                     00B7   219 ; OUTPUT PARAMETERS:
                     00B7   220 ;
                     0067   221 ;       CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
                     00B7   222 ;
                     00B7   223 ; IMPLICIT OUTPUTS:
                     00B7   224 ;
                     00B7   225 ;       R2,3,4,5,6 PRESERVED
                     00B7   226 ;
                     00B7   227 ; COMPLETION CODES:
                     00B7   228 ;
                     00B7   229 ;       NONE
                     00B7   230 ;
                     00B7   231 ; SIDE EFFECTS:
                     00B7   232 ;
                     00B7   233 ;       NONE
                     00B7   234 ;
                     00B7   235 ;--
                     00B7   236
                     00B7   237
                     00B7   238
                     00B7   239 COND1::
                  05 00B7   240          RSB                                 ; RETURN TO MAIN ROUTINE
                     00B8   241 COND1_CLEANUP::
                  05 00B8   242          RSB                                 ; RETURN TO MAIN ROUTINE
                     00B9   243 COND2::
                  52 D5 00B9   244          TSTL    R2                          ; FIRST COND 1 ELEMENT ??
                  15 12 00BB   245          BNEQ    COND2X                      ; NO -- NO CONFLICT
03 0000C135'EF43 D1 00BD   246          CMPL    COND2_E[R3],#PSL$C_USER ; IS PAGE-OWNER USER MODE ??
```

```
                     0B  13  00C5    247          BEQL    COND2X                          ; YES -- NO CONFLICT
     00000000'EF     00000000'EF  90  00C7    248          MOVB    ONES,CONFLICT           ; NO -- INDICATE CONFLICT BECAUSE ...
                                  00D2    249                                          ; ... CURRENT IMAGE PAGE MUST BE USER
                                  00D2    250 COND2X:
                                  05  00D2    251          RSB                             ; RETURN TO MAIN ROUTINE
                                  00D3    252 COND2_CLEANUP::
                                  05  00D3    253          RSB                             ; RETURN TO MAIN ROUTINE
                                  00D4    254 COND3::
 00000135'EF43   0000017E'EF44    D1  00D4    255          CMPL    ACMODE[R4],COND2_E[R3]  ; IS LOCKER AT LEAST AS ...
                                  00E1    256                                          ; ... PRIVILEGED AS OWNER ??
                     0B  15  00E1    257          BLEQ    COND3X                          ; YES -- THAT'S OK
     00000000'EF     00000000'EF  90  00E3    258          MOVB    ONES,CONFLICT           ; NO -- INDICATE CONFLICT
                                  00EE    259 COND3X:
                                  05  00EE    260          RSB                             ; RETURN TO MAIN ROUTINE
                                  00EF    261 COND3_CLEANUP::
                                  05  00EF    262          RSB                             ; RETURN TO MAIN ROUTINE
                                  00F0    263 COND4::
                                  05  00F0    264          RSB                             ; RETURN TO MAIN ROUTINE
                                  00F1    265 COND4_CLEANUP::
                                  05  00F1    266          RSB                             ; RETURN TO MAIN ROUTINE
                                  00F2    267 COND5::
                                  05  00F2    268          RSB                             ; RETURN TO MAIN ROUTINE
                                  00F3    269 COND5_CLEANUP::
                                  05  00F3    270          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                                   00F4          .SBTTL  FORM_CONDS
                                   00F4     ;++
                                   00F4     ; FUNCTIONAL DESCRIPTION:
                                   00F4     ;
                                   00F4     ;               FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                                   00F4     ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                                   00F4     ;
                                   00F4     ; CALLING SEQUENCE:
                                   00F4     ;
                                   00F4     ;       BSBW FORM_CONDS
                                   00F4     ;
                                   00F4     ; INPUT PARAMETERS:
                                   00F4     ;
                                   00F4     ;       NONE
                                   00F4     ;
                                   00F4     ; IMPLICIT INPUTS:
                                   00F4     ;
                                   00F4     ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                                   00F4     ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                                   00F4     ;       FOR X = 1,2,3,4,5 :
                                   00F4     ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                                   00F4     ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                                   00F4     ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                                   00F4     ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                                   00F4     ;
                                   00F4     ; OUTPUT PARAMETERS:
                                   00F4     ;
                                   00F4     ;       NONE
                                   00F4     ;
                                   00F4     ; IMPLICIT OUTPUTS:
                                   00F4     ;
                                   00F4     ;       NONE
                                   00F4     ;
                                   00F4     ; COMPLETION CODES:
                                   00F4     ;
                                   00F4     ;       NONE
                                   00F4     ;
                                   00F4     ; SIDE EFFECTS:
                                   00F4     ;
                                   00F4     ;       NONE
                                   00F4     ;
                                   00F4     ;--
                                   00F4
                                   00F4
                                   00F4
                                   00F4     FORM_CONDS::
                                   00F4             $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                                   0113                                             ; FORMAT CONDITIONS HEADER MSG
                        FEEA'  30  0113             BSBW    OUTPUT_MSG              ; ... AND PRINT IT
                 14       08   91  0116             CMPB    #COND1_C,#NULL          ; IS CONDITION 1 NULL ?
                          03   12  0119             BNEQU   10$                     ; NO -- CONTINUE
                        00D7   31  011B             BRW     FORM_CONDSX             ; YES -- SUBROUTINE IS FINISHED
                                   011E     10$:
   00000000'EF   00000018'EF   DE  011E             MOVAL   COND1_T,MSG_A           ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
 00000000'EF   0000001F'EF42   D0  0129             MOVL    COND1_TAB[R2],MSG_B     ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
            00000000'EF   08   90  0135             MOVB    #COND1_C,MSG_CTXT       ; SAVE CONDITION 1 CONTEXT FOR FAO
                                   013C             MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                     FEB5'   30  0148   329          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 1 MSG
                  14   00    91  014B   330          CMPB    #COND2_C,#NULL              ; IS CONDITION 2 NULL ?
                       03    12  014E   331          BNEQU   20$                         ; NO -- CONTINUE
                     00A2    31  0150   332          BRW     FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
                             0153   333 20$:
  00000000'EF   000000EC'EF  DE  0153   334          MOVAL   COND2_T,MSG_A               ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF   00000104'EF43  D0  015E   335          MOVL    COND2_TAB[R3],MSG_B         ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
        00000000'EF   00     90  016A   336          MOVB    #COND2_C,MSG_CTXT           ; SAVE CONDITION 2 CONTEXT FOR FAO
                             0171   337          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                     FE8C'   30  0171   338          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 2 MSG
                  14   04    91  0174   339          CMPB    #COND3_C,#NULL              ; IS CONDITION 3 NULL ?
                       03    12  0177   340          BNEQU   30$                         ; NO -- CONTINUE
                     0079    31  0179   341          BRW     FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
                             017C   342 30$:
  00000000'EF   00000145'EF  DE  017C   343          MOVAL   COND3_T,MSG_A               ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF   0000014D'EF44  D0  0187   344          MOVL    COND3_TAB[R4],MSG_B         ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
        00000000'EF   04     90  0193   345          MOVB    #COND3_C,MSG_CTXT           ; SAVE CONDITION 3 CONTEXT FOR FAO
                             019A   346          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                     FE57'   30  01A6   347          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 3 MSG
                  14   14    91  01A9   348          CMPB    #COND4_C,#NULL              ; IS CONDITION 4 NULL ?
                       47    13  01AC   349          BEQLU   FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   0000018E'EF  DE  01AE   350          MOVAL   COND4_T,MSG_A               ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF   0000018E'EF45  D0  01B9   351          MOVL    COND4_TAB[R5],MSG_B         ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
        00000000'EF   14     90  01C5   352          MOVB    #COND4_C,MSG_CTXT           ; SAVE CONDITION 4 CONTEXT FOR FAO
                             01CC   353          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                     FE31'   30  01CC   354          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 4 MSG
                  14   14    91  01CF   355          CMPB    #COND5_C,#NULL              ; IS CONDITION 5 NULL ?
                       21    13  01D2   356          BEQLU   FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   0000018F'EF  DE  01D4   357          MOVAL   COND5_T,MSG_A               ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF   0000018F'EF46  D0  01DF   358          MOVL    COND5_TAB[R6],MSG_B         ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
        00000000'EF   14     90  01EB   359          MOVB    #COND5_C,MSG_CTXT           ; SAVE CONDITION 5 CONTEXT FOR FAO
                             01F2   360          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                     FE0B'   30  01F2   361          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 5 MSG
                             01F5   362 FORM_CONDSX:
                           05  01F5   363          RSB                                 ; RETURN TO CALLER
```

```
01F6    365          .SBTTL  VERIFY
01F6    366 ;++
01F6    367 ; FUNCTIONAL DESCRIPTION:
01F6    368 ;
01F6    369 ;                   VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
01F6    370 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
01F6    371 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
01F6    372 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICES
01F6    373 ; ($LKWSET & $ULWSET). THEN, THE SUCCESSFUL OPERATION OF THE SERVICES IS
01F6    374 ; VERIFIED BY EXAMINING THE STATUS CODES RETURNED, THE VALUES FOR RETURN ARGS
01F6    375 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
01F6    376 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
01F6    377 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
01F6    378 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
01F6    379 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
01F6    380 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
01F6    381 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
01F6    382 ;
01F6    383 ; CALLING SEQUENCE:
01F6    384 ;
01F6    385 ;       BSBW VERIFY
01F6    386 ;
01F6    387 ; INPUT PARAMETERS:
01F6    388 ;
01F6    389 ;       NONE
01F6    390 ;
01F6    391 ; IMPLICIT INPUTS:
01F6    392 ;
01F6    393 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
01F6    394 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
01F6    395 ;       FOR X = 1,2,3,4,5 :
01F6    396 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
01F6    397 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
01F6    398 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
01F6    399 ;                 FOR CONDX_E.
01F6    400 ;
01F6    401 ; OUTPUT PARAMETERS:
01F6    402 ;
01F6    403 ;       NONE
01F6    404 ;
01F6    405 ; IMPLICIT OUTPUTS:
01F6    406 ;
01F6    407 ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
01F6    408 ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
01F6    409 ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
01F6    410 ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
01F6    411 ;       ERRORS.
01F6    412 ;
01F6    413 ; COMPLETION CODES:
01F6    414 ;
01F6    415 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
01F6    416 ;
01F6    417 ; SIDE EFFECTS:
01F6    418 ;
01F6    419 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
01F6    420 ;       (VIA RSB) IF ERROR ENCOUNTERED.
01F6    421 ;
```

```
                         01F6   422 ;--
                         01F6   423
                         01F6   424
                         01F6   425
                         01F6   426 VERIFY::
00000000'EF   95         01F6   427         TSTB    CFLAG                   ; SHOULD CONDITIONS BE PRINTED ?
              03         01FC   428         BEQL    5$                      ; NO -- CONTINUE
            FEF3         01FE   429         BSBW    FORM_CONDS              ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                         0201   430 5$:
                         0201   431 ;
                         0201   432 ; SET UP INADR ARGUMENT IN COND 1 TABLE BY ISSUING $EXPREG
                         0201   433 ;
              52         D5     0201   434         TSTL    R2                      ; FIRST COND 1 ELEMENT ??
              03         12     0203   435         BNEQ    10$                     ; NO -- CONTINUE
            00A4         31     0205   436         BRW     40$                     ; YES -- INADR ALREADY SET UP
                         0208   437 10$:
000000C4'EF42         7C     0208   438         CLRQ    INADR[R2]               ; CLEAR $EXPREG RETURN AREA
0000000C'EF           D4     020F   439         CLRL    REGION_ERG              ; ASSUME PROGRAM REGION
00000008'EF   01      9A     0215   440         MOVZBL  #1,PAGCNT_ERG           ; ... AND SINGLE PAGE COUNT
              02   52   D1     021C   441         CMPL    R2,#2                   ; IS IT PROGRAM REGION ??
                   07   15     021F   442         BLEQ    15$                     ; YES -- GO CHECK PAGE COUNT
0000000C'EF   01      9A     0221   443         MOVZBL  #1,REGION_ERG           ; NO -- MUST BE CONTROL REGION
                         0228   444 15$:
              01   52   D1     0228   445         CMPL    R2,#1                   ; IS IT SINGLE PAGE ??
                   0C   13     022B   446         BEQL    20$                     ; YES -- GO ESTABLISH MODE
              03   52   D1     022D   447         CMPL    R2,#3                   ; COULD STILL BE SINGLE PAGE
                   07   13     0230   448         BEQL    20$                     ; IT IS -- GO ESTABLISH MODE
00000008'EF   0A      9A     0232   449         MOVZBL  #10,PAGCNT_ERG          ; MULTIPLE PAGES
                         0239   450 20$:
                         0239   451         MODE    TO,30$,KRNL             ; GET KERNEL MODE, SO ACMODE ARG WORKS
                         025C   452         $EXPREG_S REGION=REGION_ERG, PAGCNT=PAGCNT_ERG, -
                         025C   453                   ACMODE=COND2_E[R3], RETADR=INADR[R2]
                         027D   454         MODE    FROM,30$                ; BACK TO USER MODE
                         027E   455         SS_CHECK NORMAL                 ; CHECK FOR NORMAL STATUS RETURN
                         02AC   456 40$:
                         02AC   457 ;
                         02AC   458 ; ****** SYSTEM SERVICE CALLS WHICH ARE THE SUBJECT OF THIS TEST CASE ******
                         02AC   459 ;
                         02AC   460         MODE    TO,50$,KRNL             ; GET KERNEL MODE, SO ACMODE ARG WORKS
                         02CF   461         $LKWSET_S INADR=INADR[R2], RETADR=RETADR,-
                         02CF   462                   ACMODE=ACMODE[R4]      ; ISSUE SUBJECT SERVICE
                         02EA   463         MODE    FROM,50$                ; BACK TO USER MODE
      00000000'8F        50   D1     02EB   464         CMPL    R0,#SS$_WASCLR          ; CODE RECEIVED = CODE EXPECTED ?
                         61   13     02F2   465         BEQLU   60$                     ; YES -- CONTINUE
00000000'EF   00000000'8F      D0     02F4   466         MOVL    #SS$_WASCLR,EXPV        ; NO -- LOAD UP EXPECTED AND ...
      00000000'EF        50   D0     02FF   467         MOVL    R0,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                         0306   468         ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM LKWSET>
                         0355   469 60$:
                         0355   470 ;
                         0355   471 ; COMPARE INADR AGAINST RETADR (EXCEPT LOW-ORDER 9 BITS) AND,
                         0355   472 ; IF OK, GO ISSUE $ULWSET.
                         0355   473 ;
      57   000000C4'EF42         7D     0355   474         MOVQ    INADR[R2],R7            ; GET INADR PAIR INTO REGS
        57   57   17   09   EF   035D   475         EXTZV   #9,#23,R7,R7            ; SHIFT RIGHT 9 BITS
57 00000010'EF   17   09   ED   0362   476         CMPZV   #9,#23,RETADR,R7        ; INADR = RETADR (FIRST OF PAIR) ??
                   13   12     036B   477         BNEQU   70$                     ; NO -- ERROR
      58   58   17   09   EF   036D   478         EXTZV   #9,#23,R8,R8            ; SHIFT RIGHT 9 BITS
```

```
     58   00000014'EF    17    09  ED 0372          CMPZV   #9,#23,RETADR+4,R8        ; INADR = RETADR (2ND OF PAIR) ??
                               03  12 037B          BNEQU   70$                       ; NO -- ERROR
                             0068  31 037D          BRW     80$                       ; YES -- SKIP ERROR PROCESSING
                                      0380      70$:
 00000000'EF   000000C4'EF42       7D 0380          MOVQ    INADR[R2],EXPV            ; LOAD UP EXPECTED AND ...
   00000000'EF   00000010'EF       7D 038C          MOVQ    RETADR,RECV               ; ... RECEIVED VALUES, THEN EXIT
                                      0397          ERR_EXIT QUAD,<UNEXPECTED VALUE FROM LKWSET FOR>,-
                                      0397                          < RETADR PAIR>
                                      03E8      80$:
                                      03E8      ;
                                      03E8      ; SET UP TO ISSUE $ULWSET
                                      03E8      ;
                                      03E8          MODE    TO,90$,KRNL               ; GET KERNEL MODE, SO ACMODE ARG WORKS
                                      040B          $ULWSET_S INADR=INADR[R2], RETADR=RETADR,-
                                      040B                    ACMODE=ACMODE[R4]       ; ISSUE SUBJECT SERVICE
                                      0426          MODE    FROM,90$                  ; BACK TO USER MODE
            00000000'8F        50  D1 0427          CMPL    R0,#SS$_WASSET            ; CODF RECEIVED = CODE EXPECTED ?
                               61  13 042E          BEQLU   100$                      ; YES - CONTINUE
   00000000'EF   00000000'8F       D0 0430          MOVL    #SS$_WASSET,EXPV          ; NO -- LOAD UP EXPECTED AND ...
            00000000'EF        50  D0 043B          MOVL    R0,RECV                   ; ... RECEIVED VALUES, THEN EXIT
                                      0442          ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM ULWSET>
                                      0491      100$:
                                      0491      ;
                                      0491      ; COMPARE INADR AGAINST RETADR (EXCEPT LOW-ORDER 9 BITS) AND,
                                      0491      ; IF OK, VERIFY ROUTINE IS COMPLETE, GO DO ANOTHER TEST CASE.
                                      0491      ;
            57   000000C4'EF42     7D 0491          MOVQ    INADR[R2],R7              ; GET INADR PAIR INTO REGS
              57     57    17  09  EF 0499          EXTZV   #9,#23,R7,R7              ; SHIFT RIGHT 9 BITS
     57   00000010'EF    17    09  ED 049E          CMPZV   #9,#23,RETADR,R7          ; INADR = RETADR (FIRST OF PAIR) ??
                               13  12 04A7          BNEQU   110$                      ; NO -- ERROR
              58     58    17  09  EF 04A9          EXTZV   #9,#23,R8,R8              ; SHIFT RIGHT 9 BITS
     58   00000014'EF    17    09  ED 04AE          CMPZV   #9,#23,RETADR+4,R8        ; INADR = RETADR (2ND OF PAIR) ??
                               03  12 04B7          BNEQU   110$                      ; NO -- ERROR
                             0068  31 04B9          BRW     VERIFYX                   ; YES -- SKIP ERROR PROCESSING
                                      04BC      110$:
 00000000'EF   000000C4'EF42       7D 04BC          MOVQ    INADR[R2],EXPV            ; LOAD UP EXPECTED AND ...
   00000000'EF   00000010'EF       7D 04C8          MOVQ    RETADR,RECV               ; ... RECEIVED VALUES, THEN EXIT
                                      04D3          ERR_EXIT QUAD,<UNEXPECTED VALUE FROM ULWSET FOR>,-
                                      04D3                          < RETADR PAIR>
                                      0524      VERIFYX:
                                   05 0524          RSB                               ; RETURN TO CALLER
```

```
              0525         .SBTTL  VFY_CLEANUP
              0525  ;++
              0525  ; FUNCTIONAL DESCRIPTION:
              0525  ;
              0525  ;               VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
              0525  ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
              0525  ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
              0525  ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
              0525  ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
              0525  ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
              0525  ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
              0525  ; POSSIBLY DISCOVERING A SECOND ERROR.
              0525  ;
              0525  ; CALLING SEQUENCE:
              0525  ;
              0525  ;       BSBW VFY_CLEANUP
              0525  ;
              0525  ; INPUT PARAMETERS:
              0525  ;
              0525  ;       NONE
              0525  ;
              0525  ; IMPLICIT INPUTS:
              0525  ;
              0525  ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
              0525  ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
              0525  ;       FOR X = 1,2,3,4,5 :
              0525  ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
              0525  ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
              0525  ;                   ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
              0525  ;                   FOR CONDX_E.
              0525  ;
              0525  ; OUTPUT PARAMETERS:
              0525  ;
              0525  ;       NONE
              0525  ;
              0525  ; IMPLICIT OUTPUTS:
              0525  ;
              0525  ;       NONE
              0525  ;
              0525  ; COMPLETION CODES:
              0525  ;
              0525  ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
              0525  ;
              0525  ; SIDE EFFECTS:
              0525  ;
              0525  ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
              0525  ;       (VIA RSB) IF ERROR ENCOUNTERED.
              0525  ;
              0525  ;--
              0525
              0525
              0525
              0525  VFY_CLEANUP::
    52   D5   0525          TSTL    R2                      ; FIRST COND 1 ELEMENT ??
    03   12   0527          BNEQ    10$                     ; NO -- CONTINUE
  0078   31   0529          BRW     VFY_CLEANUPX            ; YES -- NO PAGES TO DELETE
              052C  10$:
```

```
                57   00000004'EF42    7D  052C    578          MOVQ    INADR[R2],R7            ; AVOID INDEX MODE, USE REGS
                                 57    D5  0534    579          TSTL    R7                      ; DID $EXPREG GET SOME SPACE ABOVE ??
                                 03    12  0536    580          BNEQU   20$                     ; YES -- CONTINUE
                               0069    31  0538    581          BRW     VFY_CLEANUPX            ; NO -- NO PAGES TO DELETE
                                       053B    582 20$:
                                       053B    583          MODE TO,30$,KRNL                ; KRNL MODE TO DELETE ACQUIRED PAGES
                                       055E    584          $DELTVA_S INADR=INADR[R2], -
                                       055E    585                    ACMODE=COND2_E[R3]
                                       0575    586          MODE FROM,30$                   ; BACK TO USER MODE
                                       0576    587          SS_CHECK NORMAL                 ; CHECK RETURN STATUS VALUE
                                       05A4    588 VFY_CLEANUPX:
                                    05 05A4    589          RSB                             ; RETURN TO CALLER
                                       05A5    590          .END
```

```
$$$$                    = 000004DD R      04
$$$CHARS                = 0000002C
$$$CHARS1               = 00000006
$$$CHARS2               = 00000009
$$$CHARS3               = 0000000A
$$$CHARS4               = 00000004
$$$CHARS5               = 00000000
$$$COND_A               = 00000003
$$$$STRINGS             = 00000001
$$$$STRINGS2            = 00000005
$$T1                    = 00000000
$$T2                    = 00000004
ACMODE                    0000017E R      03
BYTE                    = 00000001   G
CFLAG                     ********     X  04
CHMRTN                    ********     X  04
CHM_CONT                  ********     X  04
COMP_SC                   ********     X  04
COND1                     000000B7 RG    04
COND1_C                 = 00000008
COND1_CLEANUP             000000B8 RG    04
COND1_E                   000000C4 R     03
COND1_H                   0000001E RG    03
COND1_T                   00000018 R     03
COND1_TAB                 0000001F R     03
COND2                     000000B9 RG    04
COND2X                    000000D2 R     04
COND2_C                 = 00000000
COND2_CLEANUP             000000D3 RG    04
COND2_E                   00000135 R     03
COND2_H                   00000103 RG    03
COND2_T                   000000EC R     03
COND2_TAB                 00000104 R     03
COND3                     000000D4 RG    04
COND3X                    000000EE R     04
COND3_C                 = 00000004
COND3_CLEANUP             000000EF RG    04
COND3_E                   0000017E R     03
COND3_H                   0000014C RG    03
COND3_T                   00000145 R     03
COND3_TAB                 0000014D R     03
COND4                     000000F0 RG    04
COND4_C                 = 00000014
COND4_CLEANUP             000000F1 RG    04
COND4_H                   0000018E RG    03
COND4_T                   0000018E R     03
COND4_TAB                 0000018E R     03
COND5                     000000F2 RG    04
COND5_C                 = 00000014
COND5_CLEANUP             000000F3 RG    04
COND5_H                   0000018F RG    03
COND5_T                   0000018F R     03
COND5_TAB                 0000018F R     03
CONFLICT                  ********     X  04
CTL$GL_PHD                ********     X  04
DESC                    = 00000010   G
EFLAG                     ********     X  04
EXPV                      ********     X  04
FAO_DESC                  ********     X  04
FAO_LEN                   ********     X  04
FORM_CONDS                000000F4 RG    04
FORM_CONDSX               000001F5 R     04
INADR                     000000C4 R     03
LONG                    = 00000004   G
MOD_MSG_CODE              ********     X  04
MOD_MSG_PRINT             ********     X  04
MSG1_INP_CTL              00000019 R     02
MSG3_ERR_CTL              00000039 RG    02
MSG_A                     ********     X  04
MSG_B                     ********     X  04
MSG_CTXT                  ********     X  04
MSG_DATA1                 ********     X  04
NOTARG                  = 00000000   G
NULL                    = 00000014   G
ONES                      ********     X  04
OUTPUT_MSG                ********     X  04
PAGCNT_ERG                00000008 R     03
PCV                       ********     X  04
PHD$Q_PRIVMSK           = 00000000
PRIVMASK                  00000000 R     03
PRIV_ARGS               = 00000002
PROCESS_ERR               ********     X  04
PSL$C_EXEC              = 00000001
PSL$C_KERNEL            = 00000000
PSL$C_SUPER             = 00000002
PSL$C_USER              = 00000003
QUAD                    = 00000008   G
RECV                      ********     X  04
REGION_ERG                0000000C R     03
REST_REGS                 ********     X  04
RETADR                    00000010 R     03
SAVE_REGS                 ********     X  04
SS$_NORMAL                ********     X  04
SS$_WASCLR                ********     X  04
SS$_WASSET                ********     X  04
SUCCESS                   ********     X  04
SYS$CMKRNL                ********    GX  04
SYS$DELTVA                ********    GX  04
SYS$EXPREG                ********    GX  04
SYS$FAO                   ********     X  04
SYS$LKWSET                ********    GX  04
SYS$SETPRN                ********    GX  04
SYS$SETPRV                ********    GX  04
SYS$ULWSET                ********    GX  04
TESTNUM                   ********     X  04
TEST_MOD_NAME             00000000 RG    02
TEST_MOD_NAME_D           00000009 R     02
TEST_MOD_SUCC             ********     X  04
TMD_ADDR                  ********     X  04
TM_CLEANUP                000000B3 RG    04
TM_SETUP                  00000000 RG    04
VERIFY                    000001F6 RG    04
VERIFYX                   00000524 R     04
VFY_CLEANUP               00000525 RG    04
```

```
VFY_CLEANUPX                    000005A4 R      04
WORD                          = 00000002  G
WRITE_MSG2                      ********    X   04
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
RODATA	00000051 (81.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	00000190 (400.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS78	000005A5 (1445.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:00.28
Command processing	108	00:00:00.68	00:00:01.69
Pass 1	260	00:00:07.33	00:00:15.71
Symbol table sort	0	00:00:00.55	00:00:01.10
Pass 2	168	00:00:01.98	00:00:06.10
Symbol table output	18	00:00:00.11	00:00:00.31
Psect synopsis output	5	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	590	00:00:10.77	00:00:25.43

```
The working set limit was 900 pages.
38233 bytes (75 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 346 non-local and 48 local symbols.
590 source lines were read in Pass 1, producing 24 object records in Pass 2.
37 pages of virtual memory were used to define 28 macros.
```

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SHRLIB]UETP.MLB;1	9
_255DUA28:[SYS.OBJ]LIB.MLB;1	1
_255DUA28:[SYSLIB]STARLET.MLB;2	15
TOTALS (all libraries)	25

```
671 GETS were required to define 25 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:SATSSS78/OBJ=OBJ$:SATSSS78 MSRC$:SATSSS78/UPDATE=(ENH$:SATSSS78)+EXECML$/LIB+SHRLIB$:UETP/LIB
```

SATSS574
LIS

SATSS578
LIS

SATSS580
LIS

SATSS582
LIS

SATSS590
LIS

SATSUT01
LIS

SATSUT05
LIS

SATSUT04
LIS

SATSUT06
LIS

SATSS581
LIS

SATSS591
LIS

SATSS579
LIS

SATSS583
LIS