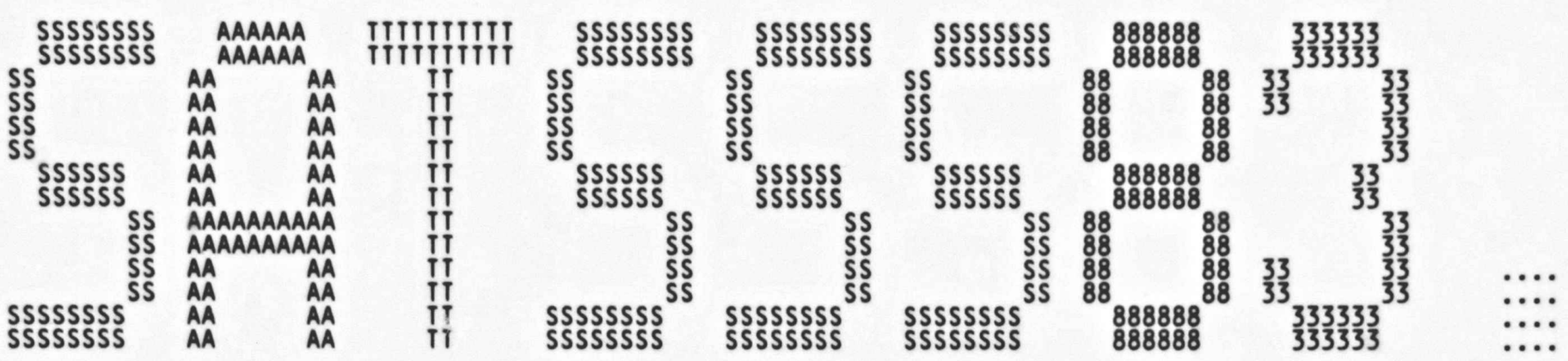
```
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
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  EEEEEEEEEEE            TTT        PPPPPPPPPPPP      SSSSSSSSS             YYY
UUU         UUU  EEEEEEEEEEE            TTT        PPPPPPPPPPPP      SSSSSSSSS             YYY
UUU         UUU  EEEEEEEEEEE            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
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
UUUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE      TTT        PPP                 SSSSSSSSSSS          YYY
```

FILEID**SATSSS83



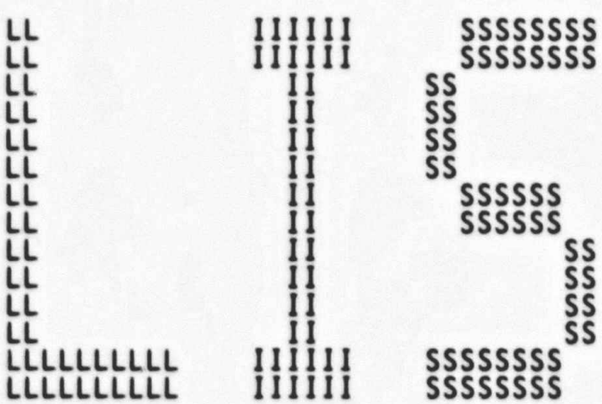


Table of contents

```
0000     1          .TITLE  SATSSS83 SATS SYSTEM SERVICE TESTS  $SETSWM (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 SATSSS83 TO TEST SUCCESSFUL
0000    36 ; OPERATION OF THE $SETSWM SYSTEM SERVICE. THE SERVICE IS INVOKED
0000    37 ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38 ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39 ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40 ; CHECKING FOR AN SS$_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: JUL, 1977
0000    47 ;
0000    48 ; MODIFIED BY:
0000    49 ;
0000    50 ;       V03-001 KDM0002        Kathleen D. Morse       28-Jun-1982
0000    51 ;                Added $SSDEF.
0000    52 ;
0000    53 ; 01    -
0000    54 ;--
```

```
0000     56          .SBTTL  DECLARATIONS
0000     57 ;
0000     58 ; INCLUDE FILES:
0000     59 ;
0000     60          $PRVDEF                         ; PRIVILEGE BIT DEFINITIONS
0000     61          $PHDDEF                         ; PROCESS HEADER OFFSETS
0000     62          $SSDEF                          ; SYSTEM STATUS CODES
0000     63 ;
0000     64 ; MACROS:
0000     65 ;
0000     66 ;
0000     67 ; EQUATED SYMBOLS:
0000     68 ;
0000     69 ;
0000     70 ; OWN STORAGE:
0000     71 ;
```

```
00000000     73          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
     0000    74 TEST_MOD_NAME:: STRING  C,<SATSSS83>     ; TEST MODULE NAME
     0009    75 TEST_MOD_NAME_D: STRING I,<SATSSS83>     ; TEST MODULE NAME DESCRIPTOR
     0019    76 MSG1_INP_CTL:   STRING  I,<  SSSSM!4ZW: CONDITIONS:>
     0039    77                                          ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
     0039    78 MSG3_ERR_CTL::  STRING  I,< *SSSSM!4ZW: !AS>
     0051    79                                          ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
```

```
         00000000    81              .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008     0000    82 PRIVMASK:            .BLKQ   1                 ; ADDR OF PRIVILEGE MASK (IN PHD)
   09 01     0008    83 STATCODES:           .BYTE   SS$_WASCLR, -     ; EXPECTED
             000A    84                              SS$_WASSET        ; ... STATUS CODES
```

```
                        000A     86          .SBTTL  CONDITION TABLES
                        000A     87 ;
                        000A     88 ;       ***** CONDITION TABLES FOR SETSWM SYSTEM SERVICE *****
                        000A     89 ;
                        000A     90          COND    1,LONG,<SWPFLG>,-
                        000A     91                    <SWAPPING DISABLED>,-
                        000A     92                    <SWAPPING ENABLED>,-
                        000A     93
00000000                003D     94                      .LONG       0       ; DISABLED
00000001                0041     95                      .LONG       1       ; ENABLED
                        0045     96 ;
                        0045     97          COND    2,NOTARG,<PREVIOUS SWAP MODE>,-
                        0045     98                    <SWAPPING PREVIOUSLY DISABLED>,-
                        0045     99                    <SWAPPING PREVIOUSLY ENABLED>,-
                        0045    100
00000000                009A    101                      .LONG       0       ; DISABLED
00000001                009E    102                      .LONG       1       ; ENABLED
                        00A2    103 ;
                        00A2    104          COND    3,NULL
                        00A3    105
                        00A3    106          COND    4,NULL
                        00A4    107
                        00A4    108          COND    5,NULL
                        00A5    109
                    00000000    110          .PSECT  SATSSS83,RD,WRT,EXE
```

```
                              0000          .SBTTL  TM_SETUP, TM_CLEANUP
                              0000     ;++
                              0000     ; FUNCTIONAL DESCRIPTION:
                              0000     ;
                              0000     ;               TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                              0000     ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                              0000     ; TEST MODULE EXECUTION.
                              0000     ;
                              0000     ; CALLING SEQUENCE:
                              0000     ;
                              0000     ;       BSBW TM_SETUP   BSBW TM_CLEANUP
                              0000     ;
                              0000     ; INPUT PARAMETERS:
                              0000     ;
                              0000     ;       NONE
                              0000     ;
                              0000     ; IMPLICIT INPUTS:
                              0000     ;
                              0000     ;       NONE
                              0000     ;
                              0000     ; OUTPUT PARAMETERS:
                              0000     ;
                              0000     ;       NONE
                              0000     ;
                              0000     ; IMPLICIT OUTPUTS:
                              0000     ;
                              0000     ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                              0000     ;                  ALL PRIVILEGES ACQUIRED.
                              0000     ;
                              0000     ; COMPLETION CODES:
                              0000     ;
                              0000     ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                              0000     ;
                              0000     ; SIDE EFFECTS:
                              0000     ;
                              0000     ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                              0000     ;       (VIA RSB) IF ERROR ENCOUNTERED.
                              0000     ;
                              0000     ;--
                              0000
                              0000
                              0000
                              0000     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,5$,KRNL                  ; KERNEL MODE TO ACCESS PHD
         59   00000000'9F D0  0048          MOVL    @#CTL$GL_PHD,R9             ; GET PROCESS HEADER ADDRESS
      00000000'EF      69 DE  004F          MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                              0056          MODE    FROM,5$ ; BACK TO USER MODE
                              0057          PRIV    ADD,ALL                     ; GET ALL PRIVILEGES
```

```
                     0077   168          $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                     0084   169          SS_CHECK  NORMAL                ; CHECK STATUS CODE RETURNED FROM SETPRN
                  05 00AE   170          RSB                             ; RETURN TO MAIN ROUTINE
                     00AF   171 TM_CLEANUP::
              FF4E'  30 00AF   172          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE END MSG
                  05 00B2   173          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                     00B3   175          .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
                     00B3   176 ;++
                     00B3   177 ; FUNCTIONAL DESCRIPTION:
                     00B3   178 ;
                     00B3   179 ;                 CONDX AND CONDX_CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
                     00B3   180 ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
                     00B3   181 ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
                     00B3   182 ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
                     00B3   183 ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
                     00B3   184 ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
                     00B3   185 ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
                     00B3   186 ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
                     00B3   187 ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
                     00B3   188 ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
                     00B3   189 ;
                     00B3   190 ; CALLING SEQUENCE:
                     00B3   191 ;
                     00B3   192 ;       BSBW CONDX   BSBW CONDX_CLEANUP
                     00B3   193 ;         WHERE X = 1,2,3,4,5
                     00B3   194 ;
                     00B3   195 ; INPUT PARAMETERS:
                     00B3   196 ;
                     00B3   197 ;       CONFLICT = 0
                     00B3   198 ;
                     00B3   199 ; IMPLICIT INPUTS:
                     00B3   200 ;
                     00B3   201 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                     00B3   202 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                     00B3   203 ;
                     00B3   204 ; OUTPUT PARAMETERS:
                     00B3   205 ;
                     00B3   206 ;       CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
                     00B3   207 ;
                     00B3   208 ; IMPLICIT OUTPUTS:
                     00B3   209 ;
                     00B3   210 ;       R2,3,4,5,6 PRESERVED
                     00B3   211 ;
                     00B3   212 ; COMPLETION CODES:
                     00B3   213 ;
                     00B3   214 ;       NONE
                     00B3   215 ;
                     00B3   216 ; SIDE EFFECTS:
                     00B3   217 ;
                     00B3   218 ;       NONE
                     00B3   219 ;
                     00B3   220 ;--
                     00B3   221
                     00B3   222
                     00B3   223
                     00B3   224 COND1::
                05   00B3   225          RSB                                ; RETURN TO MAIN ROUTINE
                     00B4   226 COND1_CLEANUP::
                05   00B4   227          RSB                                ; RETURN TO MAIN ROUTINE
                     00B5   228 COND2::
                05   00B5   229          RSB                                ; RETURN TO MAIN ROUTINE
                     00B6   230 COND2_CLEANUP::
                05   00B6   231          RSB                                ; RETURN TO MAIN ROUTINE
```

```
                00B7   232 COND3::
 05             00B7   233          RSB                     ; RETURN TO MAIN ROUTINE
                00B8   234 COND3_CLEANUP::
 05             00B8   235          RSB                     ; RETURN TO MAIN ROUTINE
                00B9   236 COND4::
 05             00B9   237          RSB                     ; RETURN TO MAIN ROUTINE
                00BA   238 COND4_CLEANUP::
 05             00BA   239          RSB                     ; RETURN TO MAIN ROUTINE
                00BB   240 COND5::
 05             00BB   241          RSB                     ; RETURN TO MAIN ROUTINE
                00BC   242 COND5_CLEANUP::
 05             00BC   243          RSB                     ; RETURN TO MAIN ROUTINE
```

```
                        00BD    245          .SBTTL  FORM_CONDS
                        00BD    246 ;++
                        00BD    247 ; FUNCTIONAL DESCRIPTION:
                        00BD    248 ;
                        00BD    249 ;               FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                        00BD    250 ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                        00BD    251 ;
                        00BD    252 ; CALLING SEQUENCE:
                        00BD    253 ;
                        00BD    254 ;       BSBW FORM_CONDS
                        00BD    255 ;
                        00BD    256 ; INPUT PARAMETERS:
                        00BD    257 ;
                        00BD    258 ;       NONE
                        00BD    259 ;
                        00BD    260 ; IMPLICIT INPUTS:
                        00BD    261 ;
                        00BD    262 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                        00BD    263 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                        00BD    264 ;       FOR X = 1,2,3,4,5 :
                        00BD    265 ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                        00BD    266 ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                        00BD    267 ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                        00BD    268 ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                        00BD    269 ;
                        00BD    270 ; OUTPUT PARAMETERS:
                        00BD    271 ;
                        00BD    272 ;       NONE
                        00BD    273 ;
                        00BD    274 ; IMPLICIT OUTPUTS:
                        00BD    275 ;
                        00BD    276 ;       NONE
                        00BD    277 ;
                        00BD    278 ; COMPLETION CODES:
                        00BD    279 ;
                        00BD    280 ;       NONE
                        00BD    281 ;
                        00BD    282 ; SIDE EFFECTS:
                        00BD    283 ;
                        00BD    284 ;       NONE
                        00BD    285 ;
                        00BD    286 ;--
                        00BD    287
                        00BD    288
                        00BD    289
                        00BD    290 FORM_CONDS::
                        00BD    291         $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                        00DC    292                                         ; FORMAT CONDITIONS HEADER MSG
                FF21'   30      00DC    293         BSBW    OUTPUT_MSG      ; ... AND PRINT IT
             14   04    91      00DF    294         CMPB    #COND1_C,#NULL  ; IS CONDITION 1 NULL ?
                  03    12      00E2    295         BNEQU   10$             ; NO -- CONTINUE
                00CB    31      00E4    296         BRW     FORM_CONDSX     ; YES -- SUBROUTINE IS FINISHED
                        00E7    297 10$:
  00000000'EF    0000000A'EF    DE      00E7    298         MOVAL   COND1_T,MSG_A           ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF    00000012'EF42    D0      00F2    299         MOVL    COND1_TAB[R2],MSG_B     ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
          00000000'EF    04    90      00FE    300         MOVB    #COND1_C,MSG_CTXT       ; SAVE CONDITION 1 CONTEXT FOR FAO
                                        0105    301         MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                     FEEC'  30  0111   302          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 1 MSG
                 14    00   91  0114   303          CMPB    #COND2_C,#NULL                  ; IS CONDITION 2 NULL ?
                       03   12  0117   304          BNEQU   20$                             ; NO -- CONTINUE
                     0096   31  0119   305          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                            011C   306 20$:
  00000000'EF   00000045'EF   DE  011C   307          MOVAL   COND2_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF   00000059'EF43   D0  0127   308          MOVL    COND2_TAB[R3],MSG_B             ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
         00000000'EF    00   90  0133   309          MOVB    #COND2_C,MSG_CTXT               ; SAVE CONDITION 2 CONTEXT FOR FAO
                            013A   310          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                     FEC3'  30  013A   311          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 2 MSG
                 14    14   91  013D   312          CMPB    #COND3_C,#NULL                  ; IS CONDITION 3 NULL ?
                       03   12  0140   313          BNEQU   30$                             ; NO -- CONTINUE
                     006D   31  0142   314          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                            0145   315 30$:
  00000000'EF   000000A2'EF   DE  0145   316          MOVAL   COND3_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF   000000A2'EF44   D0  0150   317          MOVL    COND3_TAB[R4],MSG_B             ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
         00000000'EF    14   90  015C   318          MOVB    #COND3_C,MSG_CTXT               ; SAVE CONDITION 3 CONTEXT FOR FAO
                            0163   319          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                     FE9A'  30  0163   320          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 3 MSG
                 14    14   91  0166   321          CMPB    #COND4_C,#NULL                  ; IS CONDITION 4 NULL ?
                       47   13  0169   322          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   000000A3'EF   DE  016B   323          MOVAL   COND4_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF   000000A3'EF45   D0  0176   324          MOVL    COND4_TAB[R5],MSG_B             ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
         00000000'EF    14   90  0182   325          MOVB    #COND4_C,MSG_CTXT               ; SAVE CONDITION 4 CONTEXT FOR FAO
                            0189   326          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                     FE74'  30  0189   327          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 4 MSG
                 14    14   91  018C   328          CMPB    #COND5_C,#NULL                  ; IS CONDITION 5 NULL ?
                       21   13  018F   329          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   000000A4'EF   DE  0191   330          MOVAL   COND5_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF   000000A4'EF46   D0  019C   331          MOVL    COND5_TAB[R6],MSG_B             ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
         00000000'EF    14   90  01A8   332          MOVB    #COND5_C,MSG_CTXT               ; SAVE CONDITION 5 CONTEXT FOR FAO
                            01AF   333          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                     FE4E'  30  01AF   334          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 5 MSG
                            01B2   335 FORM_CONDSX:
                       05   01B2   336          RSB                                     ; RETURN TO CALLER
```

```
01B3    338          .SBTTL  VERIFY
01B3    339 ;++
01B3    340 ; FUNCTIONAL DESCRIPTION:
01B3    341 ;
01B3    342 ;               VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
01B3    343 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
01B3    344 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
01B3    345 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
01B3    346 ; ($SETSWM). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
01B3    347 ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
01B3    348 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
01B3    349 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
01B3    350 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
01B3    351 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
01B3    352 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
01B3    353 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
01B3    354 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
01B3    355 ;
01B3    356 ; CALLING SEQUENCE:
01B3    357 ;
01B3    358 ;       BSBW VERIFY
01B3    359 ;
01B3    360 ; INPUT PARAMETERS:
01B3    361 ;
01B3    362 ;       NONE
01B3    363 ;
01B3    364 ; IMPLICIT INPUTS:
01B3    365 ;
01B3    366 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
01B3    367 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
01B3    368 ;       FOR X = 1,2,3,4,5 :
01B3    369 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
01B3    370 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
01B3    371 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
01B3    372 ;                 FOR CONDX_E.
01B3    373 ;
01B3    374 ; OUTPUT PARAMETERS:
01B3    375 ;
01B3    376 ;       NONE
01B3    377 ;
01B3    378 ; IMPLICIT OUTPUTS:
01B3    379 ;
01B3    380 ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
01B3    381 ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
01B3    382 ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
01B3    383 ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
01B3    384 ;       ERRORS.
01B3    385 ;
01B3    386 ; COMPLETION CODES:
01B3    387 ;
01B3    388 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
01B3    389 ;
01B3    390 ; SIDE EFFECTS:
01B3    391 ;
01B3    392 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
01B3    393 ;       (VIA RSB) IF ERROR ENCOUNTERED.
01B3    394 ;
```

```
                                01B3   395 ;--
                                01B3   396
                                01B3   397
                                01B3   398
                                01B3   399 VERIFY::
            00000000'EF   95    01B3   400         TSTB    CFLAG                   ; SHOULD CONDITIONS BE PRINTED ?
                      03   13    01B9   401         BEQL    5$                      ; NO -- CONTINUE
                    FEFF   30    01BB   402         BSBW    FORM_CONDS              ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                                01BE   403 5$:
                                01BE   404         $SETSWM_S COND2_E[R3]           ; ISSU PRELIM SERV TO ESTAB "PREV" CONDITION
                                01CC   405 ;
                                01CC   406 ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                                01CC   407 ;
                                01CC   408         $SETSWM_S SWPFLG[R2]            ; ISSUE SUBJECT SERVICE
     59   00000008'EF43   9A    01DA   409         MOVZBL  STATCODES[R3],R9        ; GET EXPECTED STATUS CODE
                   59   50    D1    01E2   410         CMPL    R0,R9               ; CODE RECEIVED = CODE EXPECTED ?
                      03   12    01E5   411         BNEQ    10$                     ; NO -- GO PROCESS ERROR
                    005B   31    01E7   412         BRW     VERIFYX                 ; YES -- ALL FINISHED
                                01EA   413 10$:
          00000000'EF   59    D0    01EA   414         MOVL    R9,EXPV             ; LOAD UP EXPECTED AND
          00000000'EF   50    D0    01F1   415         MOVL    R0,RECV             ; ... RECEIVED VALUES, THEN EXIT
                                01F8   416         ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED BY SETSWM>
                                0245   417 VERIFYX:
                                0245   418         RSB                             ; RETURN TO CALLER
```

```
                        .SBTTL  VFY_CLEANUP
                ;++
                ; FUNCTIONAL DESCRIPTION:
                ;
                ;               VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
                ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
                ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
                ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
                ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
                ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
                ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
                ; POSSIBLY DISCOVERING A SECOND ERROR.
                ;
                ; CALLING SEQUENCE:
                ;
                ;       BSBW VFY_CLEANUP
                ;
                ; INPUT PARAMETERS:
                ;
                ;       NONE
                ;
                ; IMPLICIT INPUTS:
                ;
                ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                ;       FOR X = 1,2,3,4,5 :
                ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
                ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
                ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
                ;                 FOR CONDX_E.
                ;
                ; OUTPUT PARAMETERS:
                ;
                ;       NONE
                ;
                ; IMPLICIT OUTPUTS:
                ;
                ;       NONE
                ;
                ; 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.
                ;
                ;--


                VFY_CLEANUP::
                        $SETSWM_S  #1                   ; MAKE SURE SWAPPING IS ENABLED
05              RSB                                     ; RETURN TO CALLER
                        .END
```

```
$$$$                    = 00000202 R     04
$$$CHARS                = 00000028
$$$CHARS1               = 0000001C
$$$CHARS2               = 0000001B
$$$CHARS3               = 00000000
$$$CHARS4               = 00000000
$$$CHARS5               = 00000000
$$$COND_A               = 00000001
$$$STRINGS              = 00000001
$$$STRINGS2             = 00000005
$$T2                    = 00000004
BYTE                    = 00000001  G
CFLAG                     ********    X   04
CHMRTN                    ********    X   04
CHM_CONT                  ********    X   04
COMP_SC                   ********    X   04
CONDT                     000000B3 RG     04
COND1_C                 = 00000004
COND1_CLEANUP             000000B4 RG     04
COND1_E                   0000003D R      03
COND1_H                   00000011 RG     03
COND1_T                   0000000A R      03
COND1_TAB                 00000012 R      03
COND2                     000000B5 RG     04
COND2_C                 = 00000000
COND2_CLEANUP             000000B6 RG     04
COND2_E                   0000009A R      03
COND2_H                   00000058 RG     03
COND2_T                   00000045 R      03
COND2_TAB                 00000059 R      03
COND3                     000000B7 RG     04
COND3_C                 = 00000014
COND3_CLEANUP             000000B8 RG     04
COND3_H                   000000A2 RG     03
COND3_T                   000000A2 R      03
COND3_TAB                 000000A2 R      03
COND4                     000000B9 RG     04
COND4_C                 = 00000014
COND4_CLEANUP             000000BA RG     04
COND4_H                   000000A3 RG     03
COND4_T                   000000A3 R      03
COND4_TAB                 000000A3 R      03
COND5                     000000BB RG     04
COND5_C                 = 00000014
COND5_CLEANUP             000000BC RG     04
COND5_H                   000000A4 RG     03
COND5_T                   000000A4 R      03
COND5_TAB                 000000A4 R      03
CTL$GL_PHD                ********      X 04
DESC                    = 00000010  G
EFLAG                     ********    X   04
EXPV                      ********    X   04
FAO_DESC                  ********    X   04
FAO_LEN                   ********    X   04
FORM_CONDS                000000BD RG     04
FORM_CONDSX               000001B2 R      04
LONG                    = 00000004  G
MOD_MSG_CODE              ********    X   04
MOD_MSG_PRINT             ********    X   04
MSGT_INP_CTL              00000019 R      02
MSG3_ERR_CTL              00000039 RG     02
MSG_A                     ********    X   04
MSG_B                     ********    X   04
MSG_CTXT                  ********    X   04
MSG_DATA1                 ********    X   04
NOTARG                  = 00000000  G
NULL                    = 00000014  G
OUTPUT_MSG                ********    X   04
PCV                       ********    X   04
PHD$Q_PRIVMSK           = 00000000
PRIVMASK                  00000000 R      03
PRIV_ARGS               = 00000002
PROCESS_ERR               ********    X   04
QUAD                    = 00000008  G
RECV                      ********    X   04
REST_REGS                 ********    X   04
SAVE_REGS                 ********    X   04
SS$_NORMAL              = 00000001
SS$_WASCLR              = 00000001
SS$_WASSET              = 00000009
STATCODES                 00000008 R      03
SUCCESS                   ********    X   04
SWPFLG                    0000003D R      03
SYS$CMKRNL                ********   GX   04
SYS$FAO                   ********    X   04
SYS$SETPRN                ********   GX   04
SYS$SETPRV                ********   GX   04
SYS$SETSWM                ********   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                000000AF RG     04
TM_SETUP                  00000000 RG     04
VERIFY                    000001B3 RG     04
VERIFYX                   00000245 R      04
VFY_CLEANUP               00000246 RG     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	000000A5 (165.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS83	00000250 (592.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.08	00:00:00.65
Command processing	133	00:00:00.69	00:00:03.39
Pass 1	282	00:00:08.42	00:00:15.16
Symbol table sort	0	00:00:01.08	00:00:01.19
Pass 2	98	00:00:01.81	00:00:04.76
Symbol table output	12	00:00:00.13	00:00:00.36
Psect synopsis output	3	00:00:00.03	00:00:00.04
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	565	00:00:12.24	00:00:25.55

The working set limit was 1500 pages.
44664 bytes (88 pages) of virtual memory were used to buffer the intermediate code.
There were 40 pages of symbol table space allocated to hold 698 non-local and 14 local symbols.
475 source lines were read in Pass 1, producing 20 object records in Pass 2.
33 pages of virtual memory were used to define 24 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	11
TOTALS (all libraries)	21

1005 GETS were required to define 21 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:SATSSS83/OBJ=OBJ$:SATSSS83 MSRC$:SATSSS83/UPDATE=(ENH$:SATSSS83)+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