```
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
```

FILEID**SATSSF16

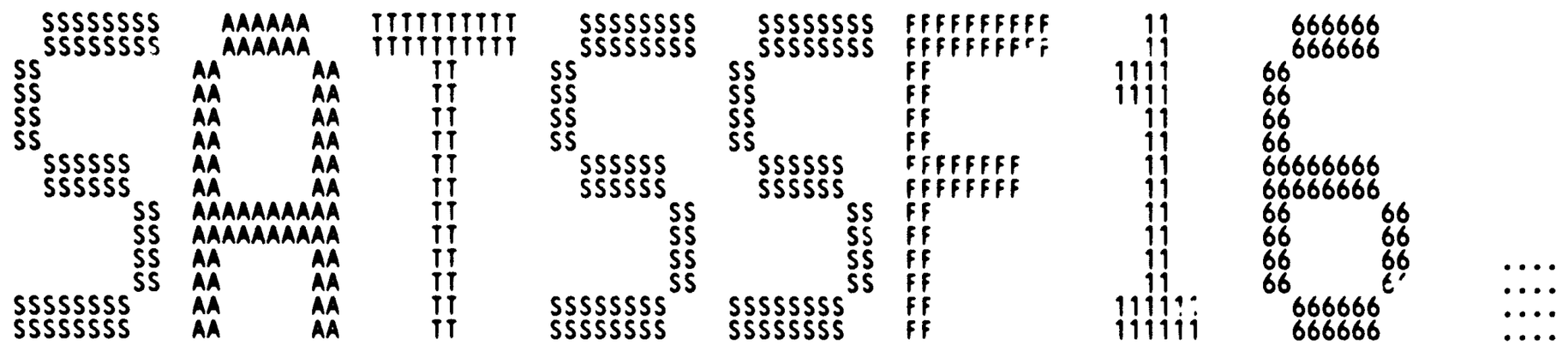

Table of contents

```
0000     1          .TITLE  SATSSF16 - SATS SYSTEM SERVICE TESTS  (FAILING 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:     SATS SYSTEM SERVICE TESTS
0000    31  ;
0000    32  ; ABSTRACT:     THE SATSSF16 MODULE TESTS THE EXECUTION OF CERTAIN
0000    33  ; VMS SYSTEM SERVICES, INVOKED IN SUCH A WAY AS TO EXPECT FAILING
0000    34  ; STATUS CODES. THE SYSTEM SERVICES TESTED AND THE STATUS CODES
0000    35  ; EXPECTED ARE SUMMARIZED AS ARGUMENTS TO THE TESTSERV MACROS
0000    36  ; WHICH APPEAR NEAR THE END OF THIS LISTING. SUCCESSFUL STATUS
0000    37  ; CODES ARE TESTED IN OTHER MODULES.
0000    38  ;
0000    39  ;
0000    40  ; ENVIRONMENT:  USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000    41  ;                DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000    42  ;
0000    43  ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: MMM, 1978
0000    44  ;         PAUL D. FAY (DISPSERV & TESTSERV MACROS)
0000    45  ;
0000    46  ; MODIFIED BY:
0000    47  ;
0000    48  ;          , : VERSION
0000    49  ; 01     -
0000    50  ;--
```

```
                     0000     52          .SBTTL  DECLARATIONS
                     0000     53 ;
                     0000     54 ; INCLUDE FILES:
                     0000     55 ;
                     0000     56          $PHDDEF                             ; PROCESS HEADER OFFSET SYMBOLS
                     0000     57          $PCBDEF                             ; PROCESS CONTROL BLOCK OFFSET SYMBS
                     0000     58          $STSDEF                             ; STATUS MESSAGE SYMBOLS
                     0000     59          $PRVDEF                             ; SYMBOL DEFS FOR PRIVILEGES
                     0000     60          $UETPDEF                            ; UETP MSG CODE DEFINITIONS
                     0000     61          $SHR_MESSAGES UETP,116,<<TEXT,INFO>>
                     0000     62                                              ; DEFINE UETP$_TEXT
                     0000     63                                              ; GET RID OF MACRO DEFINITIONS
                     0000     64 ;
                     0000     65 ; MACROS:
                     0000     66 ;
                     0000     67 ;
                     0000     68 ; EQUATED SYMBOLS:
                     0000     69 ;
00000000             0000     70 WARNING          = 0                         ; WARNING SEVERITY VALUE FOR MSGS
00000001             0000     71 SUCCESS          = 1                         ; SUCCESS SEVERITY VALUE FOR MSGS
00000002             0000     72 ERROR            = 2                         ; ERROR SEVERITY VALUE FOR MSGS
00000003             0000     73 INFO             = 3                         ; INFORMATIONAL SEV VALUE FOR MSGS
00000004             0000     74 SEVERE           = 4                         ; SEVERE (FATAL) SEV VALUE FOR MSGS
00000000             0000     75 TCG_NO           =  0                        ; INITIALIZE TEST CASE GROUP NUMBER
00000000             0000     76 GRP_TOTAL        =  0                        ; INITIALIZE TEST CASE GROUP TOTAL
00007FFF             0000     77 R0_THRU_SP       = ^M<R0,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,AP,FP,SP>
                     0000     78 ;
                     0000     79 ; OWN STORAGE:
                     0000     80 ;
```

```
                    00000000     82          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
                  BFFC  0000     83 REG_COMP_MASK:  .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,AP,FP> ! ^X8000 -
                        0002     84                                          ; REG COMPARE MASK (HIGH-ORDER ...
                        0002     85                                          ; ... BIT MUST BE ON
                        0002     86 ERR_MSG_FAOCTL: STRING  I,<!/!AC!1ZB!1ZB: REGISTER !2UW CONTENTS ALTERED>, -
                        0002     87                 <;  BEFORE SERVICE CALL: !8XL   AFTER SERVICE CALL: !8XL>
                        006E     88 TEST_MOD_NAME:  STRING  C,<SATSSF16>     ; TEST MODULE NAME
                        0077     89 TEST_MOD_BEG:   STRING  C,<begun>        ; DISPOSITION FIELD OF TEST MOD MSG
                        007D     90 TEST_MOD_SUCC:  STRING  C,<successful>   ; DISPOSITION FIELD OF TEST MOD MSG
                        0088     91 TEST_MOD_FAIL:  STRING  C,<failed>       ; DISPOSITION FIELD OF TEST MOD MSG
                        008F     92 TEST_MOD_NAME_D: STRING I,<SATSSF16>     ; TEST MODULE NAME DESCRIPTOR
                        009F     93 TTNAME:         STRING  I,<TT>           ; TERMINAL LOGICAL NAME
00000000'00000000'      00A9     94 INADR:          .LONG   NOACCESS,NOACCESS ; PAGE ADDRESS OF NOACCESS PSECT
            00000000'   00B1     95 PROT:           .LONG   PRT$C_NA         ; PROTECTION CODE FOR NOACCESS PSECT
FFFFFFFF FFFFFFFF       00B5     96 ONES:           .LONG   -1,-1            ; A QUADWORD OF 1-BITS
                        00BD     97 CTRSTR_FAO:                              ; CTRSTR ARGUMENT FOR FAO
                        00BD     98 CTRSTR_FAL:                              ; CTRSTR ARGUMENT FOR FAOL
                        00BD     99                 STRING  I,<TEST: !AC>
                        00CE    100 CTRSTR_FAO10:                            ; CTRSTR ARGUMENT FOR FAO
                        00CE    101 CTRSTR_FAL10:                            ; CTRSTR ARGUMENT FOR FAOL
                        00CE    102                STRING  I,<TEST: !8C>
                000000E3'       00DF    103 P1_FAO:        .ADDRESS COUNTED_STR     ; P1 ARGUMENT FOR FAO
                        00E3    104 COUNTED_STR:   STRING  C,<A VALID COUNTED STRING>
                00000000        00FA    105 P2_FAO:        .LONG   0                ; P2 ARGUMENT FOR FAO
                000000DF'       00FE    106 PRMLST_FAL:    .ADDRESS P1_FAO          ; PRMLST ARGUMENT FOR FAOL
                        0102    107 MSGBUF_SNE:
                        0102    108 MSGBUF_SNE10:  STRING  I,<SATSSF16 ERROR LOG BUFFER> ; MSGBUF ARG FOR SNDERR
                        0123    109 MY_DISK:       STRING  I,<SYS$DISK>     ; LOGICAL NAME FOR USER DISK
```

```
      00000000   111          .PSECT  RWDATA,RD,WRT,NOEXE
00000004  0000   112 TPID:             .BLKL   1               ; PROCESS ID FOR THIS PROCESS
00000008  0004   113 CURRENT_TC:       .BLKL   1               ; PTR TO CURRENT TEST CASE
00000044  0008   114 REG_SAVE_AREA:    .BLKL   15              ; SAVE AREA FOR ALL REGS (SANS PC)
007480D9  0044   115 MOD_MSG_CODE:     .LONG   UETP$_SATSMS    ; TEST MODULE MSG CODE FOR PUTMSG
0000004C  0048   116 CLOB_REG_NO:      .BLKL   1               ; CLOBBERED REG NO (FOR FAO ERR MSG)
00000050  004C   117 REG_BEFORE_SS:    .BLKL   1               ; REG CONTENTS BEFORE S.S.
          0050   118                                           ; ... (FOR FAO ERROR MSG)
00000054  0050   119 REG_AFTER_SS:     .BLKL   1               ; REG CONTENTS AFTER S.S.
          0054   120                                           ; ... (FOR FAO ERROR MSG)
          0054   121 $$TSTN$$:         STRING  C,<  SF   >     ; ASCII PORTION OF TEST CASE NAME
0000006E' 005C   122 TMN_ADDR:         .ADDRESS TEST_MOD_NAME  ; ADDR OF TEST MOD NAME FOR FAO
00000077' 0060   123 TMD_ADDR:         .ADDRESS TEST_MOD_BEG   ; ADDR OF T.M. DISP FIELD FOR FAO
00000068  0064   124 TS_EP:            .BLKL   1               ; ENTRY PNT FOR CURR TESTSERV MACRO
00000070  0068   125 RETADR:           .BLKL   2               ; RETURN LONGWORDS FOR SETPRT
00000071  0070   126 PRVPRT:           .BLKB   1               ; PROT RETURN BYTE FOR SETPRT
00000079  0071   127 PRIVMASK:         .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000007D  0079   128 CHM_CONT:         .BLKL   1               ; CHANGE MODE CONTINUE ADDRESS
00000091  007D   129 REGS:             .BLKL   5               ; AREA FOR COND INDEX REGS (R2-R6)
          0091   130 OUTLEN_FAO:                               ; OUTLEN ARGUMENT FOR FAO
          0091   131 OUTLEN_FAL:                               ; OUTLEN ARGUMENT FOR FAOL
00000093  0091   132                  .BLKW   1
          0093   133 OUTBUF_FAO:                               ; OUTBUF ARGUMENT FOR FAO
          0093   134 OUTBUF_FAL:                               ; OUTBUF ARGUMENT FOR FAOL
          0093   135                  STRING  0,50
          00CD   136 OUTBUF_FAO30:                             ; OUTBUF ARGUMENT FOR FAO
          00CD   137 OUTBUF_FAL30:                             ; OUTBUF ARGUMENT FOR FAOL
          00CD   138                  STRING  0,1
```

```
                    140          .PSECT  SATS_ACCVIO_1,RD,WRT,NOEXE,PAGE
00000200  0000    141 EMPTY:          .BLKB   512             , RESERVE A PAGE OF SPACE
          0200    142 ;
          0200    143 ; +
          0200    144 ; ************************************************************************
          0200    145 ; *                                                                      *
          0200    146 ; *     THE ORDER OF STATEMENTS IN THIS PSECT IS CRITICAL.               *
          0200    147 ; *     DO NOT RE-ARRANGE THE VARIABLES. CONSULT SATS                    *
          0200    148 ; *     FUNCTIONAL SPECIFICATION FOR A DESCRIPTION OF THE USE            *
          0200    149 ; *     OF THE EMPTY PSECT (AND ITS COMPANION PSECT, NOACCESS).          *
          0200    150 ; *                                                                      *
          0200    151 ; ************************************************************************
          0200    152 ; -
          0200    153 ;
          0200    154 ; TYPE AAAAA_SSSX1 (TYPE AAAAA_SSSX2 IF NOT DESC) GO HERE:
000001F3  0200    155                 .=  . - 13              ; ALLOW ROOM FOR STRING DESCRIPTOR
          01F3    156 ; TYPE AAAAA_SSSX5 GO HERE:
00000006  01F3    157                 .LONG   6               ; STRING LENGTH (WILL CROSS PSECT BOUNDARY)
000001FB' 01F7    158                 .ADDRESS .+4            ; STRING ADDRESS
          01FB    159 ; TYPE AAAAA_SSSX3 GO HERE:
000001FC  01FB    160                 .BLKB   1               ; LOW-ORDER BYTE OF STRING LENGTH
          01FC    161 ; TYPE AAAAA_SSSX2 GO HERE:
00000200  01FC    162                 .BLKL   1               ; STRING LENGTH
          0200    163 ;
          0200    164 ;
          0200    165 ;
          0200    166 ;
00000000          167          .PSECT  SATS_ACCVIO_2,RD,WRT,NOEXE,PAGE
00000200  0000    168 NOACCESS:       .BLKB   512             ; RESERVE A PAGE OF SPACE
00000000  0200    169                 .=  . - 512             ; RETURN LOC CTR TO BEGINNING OF PSECT
00000000' 0000    170                 .ADDRESS EMPTY          ; ADDRESS OF ACCESSIBLE STRING
00000000' 0004    171                 .ADDRESS EMPTY/^X100    ; ADDRESS OF ACCESSIBLE STRING
          0008    172 ;+
          0008    173 ; *** NOTE -- DO NOT CHANGE LOCATION OR SEQUENCE OF ABOVE STATEMENTS!
          0008    174 ; ***         THIS PSECT (NOACCESS) MUST APPEAR IN MEMORY IMMEDIATELY
          0008    175 ; ***         FOLLOWING THE EMPTY PSECT. PSECT NAMES AND OPTIONS WILL BE
          0008    176 ; ***         CHOSEN TO FORCE THE DESIRED PSECT ORDERING.
          0008    177 ;-
          0008    178 ;
          0008    179 MSGBUF_SNE11:   STRING  I,<SFSNE11>     ; MSGBUF ARGUMENT FOR SNDERR
          0017    180 ;
          0017    181 ;
          0017    182 ;
00000000          183          .PSECT  SATSSF16,RD,WRT,EXE,LONG
```

```
0000    185          .SBTTL  SATSSF16
0000    186 ;++
0000    187 ; FUNCTIONAL DESCRIPTION:
0000    188 ;
0000    189 ;                   AFTER PERFORMING SOME INITIAL HOUSEKEEPING, SUCH AS
0000    190 ; PRINTING THE MODULE BEGIN MESSAGE AND ACQUIRING ALL PRIVILEGES,
0000    191 ; THE SATSSF16 ROUTINE EXECUTES THE TEST_SERV_EXEC MACRO TO RUN
0000    192 ; ALL TEST CASES. WHEN THE MACRO COMPLETES ITS EXECUTION, SATSSF16
0000    193 ; PRINTS A TEST MODULE SUCCESS OR FAIL MESSAGE AND EXITS TO THE
0000    194 ; OPERATING SYSTEM. TEST_SERV_EXEC CALLS THE TC_CONTROL/TESTSERV
0000    195 ; CO-ROUTINE PAIR ONCE PER TEST CASE GROUP TO EXECUTE ALL TEST
0000    196 ; CASES IN THAT GROUP. EACH TEST CASE GROUP IS DEFINED BY BOUNDING
0000    197 ; ITS TEST CASES WITH A TC_GROUP MACRO BEFORE THE FIRST TEST CASE
0000    198 ; AND A TCEND MACRO AFTER THE LAST ONE. THE TEST CASES THEMSELVES
0000    199 ; ARE DEFINED WITHIN THESE BOUNDS BY PRECEDING EACH WITH A
0000    200 ; NEXT_TEST_CASE MACRO. TC_CONTROL/TESTSERV EXECUTES THE CODE
0000    201 ; FOLLOWING EACH NEXT_TEST_CASE MACRO IMMEDIATELY BEFORE ISSUING
0000    202 ; THE SYSTEM SERVICE AS REQUESTED IN THE TESTSERV MACRO. TC_CONTROL/
0000    203 ; TESTSERV ALSO CHECKS THE RESULTS OF THE SERVICE WITH RESPECT
0000    204 ; TO ITS EXPECTED STATUS CODE AND PRINTS ANY REQUIRED FAILURE
0000    205 ; MESSAGES FOR THE TEST CASE. THE CODE APPEARING AFTER EACH
0000    206 ; NEXT_TEST_CASE MACRO IS MERELY TO SET UP CONDITIONS REQUIRED
0000    207 ; FOR THE SYSTEM SERVICE AND TO CLEAN UP ANY RESOURCES ACQUIRED
0000    208 ; BY THE PREVIOUS TEST CASE.
0000    209 ;
0000    210 ; CALLING SEQUENCE:
0000    211 ;
0000    212 ;       $ RUN SATSSF16  ...  (DCL COMMAND)
0000    213 ;
0000    214 ; INPUT PARAMETERS:
0000    215 ;
0000    216 ;       NONE
0000    217 ;
0000    218 ; IMPLICIT INPUTS:
0000    219 ;
0000    220 ;       NONE
0000    221 ;
0000    222 ; OUTPUT PARAMETERS:
0000    223 ;
0000    224 ;       NONE
0000    225 ;
0000    226 ; IMPLICIT OUTPUTS:
0000    227 ;
0000    228 ;       MESSAGES TO SYS$OUTPUT ARE THE ONLY OUTPUT FROM SATSSF16.
0000    229 ;       THEY ARE OF THE FORM:
0000    230 ;
0000    231 ;               %UETP-S-SATSMS, TEST MODULE SATSSF16 BEGUN ... (BEGIN MSG)
0000    232 ;               %UETP-S-SATSMS, TEST MODULE SATSSF16 SUCCESSFUL ... (END MSG)
0000    233 ;               %UETP-E-SATSMS, TEST MODULE SATSSF16 FAILED ... (END MSG)
0000    234 ;               %UETP-I-TEXT, ... (VARIABLE INFORMATION ABOUT A TEST MODULE FAILURE)
0000    235 ;
0000    236 ; COMPLETION CODES:
0000    237 ;
0000    238 ;       THE SATSSF16 ROUTINE TERMINATES WITH A $EXIT TO THE
0000    239 ;       OPERATING SYSTEM WITH A STATUS CODE DEFINED BY UETP$_SATSMS.
0000    240 ;
0000    241 ; SIDE EFFECTS:
```

```
                                0000   242 ;
                                0000   243 ;       NONE
                                0000   244 ;
                                0000   245 ;--
                                0000   246
                                0000   247
                                0000   248
                                0000   249 SATSSF16:
                          0FFC  0000   250         .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11>
                                0002   251                                         ; ENTRY MASK
                                0002   252         $WAKE_S  TPID                   ; GET PID OF THIS PROCESS
                                0011   253         $HIBER_S                        ; UNDO WAKE
                                0018   254         $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                       0AFA   30 0025   255         BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE BEGIN MSG
 00000060'EF    0000007D'EF   DE 0028   256         MOVAL   TEST_MOD_SUCC,TMD_ADDR  ; ASSUME END MSG WILL SHOW SUCCESS
00000044'EF   03   00   01   F0 0033   257         INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
                                003C   258         MODE    TO,10$,KRNL,NOREGS      ; KERNEL MODE TO ACCESS PHD
           59   00000000'9F   D0 0059   259         MOVL    @#CTL$GL_PHD,R9         ; GET PROCESS HEADER ADDRESS
        00000071'EF   69   DE 0060   260         MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                                0067   261         MODE    FROM,TO$                ; GET BACK TO USER MODE
                                0068   262         PRIV    ADD,ALL                 ; GET ALL PRIVILEGES
                                0088   263         DISPSERV                        ; SET UP DISPLAY INFO FOR TESTSERV
                                021D   264         $SETPRT_S INADR=INADR, RETADR=RETADR, -
                                021D   265                   PROT=PROT, PRVPRT=PRVPRT
                                023E   266                                         ; SET NOACCESS PSECT ...
                                023E   267                                         ; ... FOR NO USER ACCESS
                       0788   31 023E   268         BRW     EXECUTE                 ; GO EXECUTE ALL TEST CASES
                                0241   269 ;
                                0241   270         TC_GROUP        FAO,1,TS1
                                0268   271 ;
                                0268   272         NEXT_TEST_CASE  SFFA010
```

```
0268   273 ;
0268   274 ;++
0268   275 ; ************************************************************
0268   276 ; *
0268   277 ; *  TEST CASE NAME:              SFFA010
0268   278 ; *
0268   279 ; *  SYSTEM SERVICE:              FAO
0268   280 ; *
0268   281 ; *  ARGUMENT UNDER TEST:         CTRSTR_FA010
0268   282 ; *
0268   283 ; *  INPUT CONDITIONS:
0268   284 ; *     INVALID DIRECTIVE IN FAO CONTROL STRING
0268   285 ; *
0268   286 ; *  EXPECTED RESULTS:
0268   287 ; *     1) SYSTEM STATUS CODE:  BADPARAM
0268   288 ; *     2) REGISTERS R2 THROUGH FP UNCHANGED
0268   289 ; *
0268   290 ; ************************************************************
0268   291 ;--
0268   292 ;
0268   293 ;
0268   294         NEXT_TEST_CASE  SFFA030
```

```
                                F 1
0274    295 ;
0274    296 ;++
0274    297 ; ********************************************************
0274    298 ; *
0274    299 ; *  TEST CASE NAME:              SFFA030
0274    300 ; *
0274    301 ; *  SYSTEM SERVICE:              FAO
0274    302 ; *
0274    303 ; *  ARGUMENT UNDER TEST:         OUTBUF_FA030
0274    304 ; *
0274    305 ; *  INPUT CONDITIONS:
0274    306 ; *     OUTPUT BUFFER (LENGTH 1) TOO SHORT FOR
0274    307 ; *       INCOMING DATA.
0274    308 ; *
0274    309 ; *  EXPECTED RESULTS:
0274    310 ; *     1) SYSTEM STATUS CODE:  BUFFEROVF
0274    311 ; *     2) REGISTERS R2 THROUGH FP UNCHANGED
0274    312 ; *
0274    313 ; ********************************************************
0274    314 ;--
0274    315 ;
0274    316 ;
0274    317         TCEND
```

```
0275    318 ;
0275    319
0275    319         TC_GROUP        FAL,1,TS2
029C    320 ;
029C    321
029C    321         NEXT_TEST_CASE  SFFAL10
```

```
                                H 1
029C    322 ;
029C    323 ;++
029C    324 ; ******************************************************
029C    325 ; *
029C    326 ; *  TEST CASE NAME:              SFFAL10
029C    327 ; *
029C    328 ; *  SYSTEM SERVICE:              FAOL
029C    329 ; *
029C    330 ; *  ARGUMENT UNDER TEST:         CTRSTR_FAL10
029C    331 ; *
029C    332 ; *  INPUT CONDITIONS:
029C    333 ; *     INVALID DIRECTIVE IN FAO CONTROL STRING
029C    334 ; *
029C    335 ; *  EXPECTED RESULTS:
029C    336 ; *     1) SYSTEM STATUS CODE:  BADPARAM
029C    337 ; *     2) REGISTERS R2 THROUGH FP UNCHANGED
029C    338 ; *
029C    339 ; ******************************************************
029C    340 ;--
029C    341 ;
029C    342 ;
029C    343         NEXT_TEST_CASE  SFFAL30
```

```
02A8    344 ;
02A8    345 ;++
02A8    346 ; ******************************************************
02A8    347 ; *
02A8    348 ; *  TEST CASE NAME:              SFFAL30
02A8    349 ; *
02A8    350 ; *  SYSTEM SERVICE:              FAOL
02A8    351 ; *
02A8    352 ; *  ARGUMENT UNDER TEST:         OUTBUF_FAL30
02A8    353 ; *
02A8    354 ; *  INPUT CONDITIONS:
02A8    355 ; *     OUTPUT BUFFER (LENGTH 1) TOO SHORT FOR
02A8    356 ; *       INCOMING DATA.
02A8    357 ; *
02A8    358 ; *  EXPECTED RESULTS:
02A8    359 ; *     1) SYSTEM STATUS CODE:  BUFFEROVF
02A8    360 ; *     2) REGISTERS R2 THROUGH FP UNCHANGED
02A8    361 ; *
02A8    362 ; ******************************************************
02A8    363 ;--
02A8    364 ;
02A8    365 ;
02A8    366         TCEND
```

```
02A9    367 ;
02A9    368         TC_GROUP        SNE,1,TS3
02D0    369 ;
02D0    370         NEXT_TEST_CASE  SFSNE10
```

```
                                        K 1
02D0    371 ;
02D0    372 ;++
02D0    373 ; ******************************************************
02D0    374 ; *
02D0    375 ; *  TEST CASE NAME:              SFSNE10
02D0    376 ; *
02D0    377 ; *  SYSTEM SERVICE:              SNDERR
02D0    378 ; *
02D0    379 ; *  ARGUMENT UNDER TEST:         MSGBUF_SNE10
02D0    380 ; *
02D0    381 ; *  INPUT CONDITIONS:
02D0    382 ; *     USER DOES NOT HAVE THE PRIVILEGE TO SEND A
02D0    383 ; *       MESSAGE TO THE ERROR LOGGER.
02D0    384 ; *
02D0    385 ; *  EXPECTED RESULTS:
02D0    386 ; *     1) SYSTEM STATUS CODE:  NOPRIV
02D0    387 ; *     2) REGISTERS R2 THROUGH FP UNCHANGED
02D0    388 ; *
02D0    389 ; ******************************************************
02D0    390 ;--
02D0    391 ;
02D0    392         PRIV    REM,BUGCHK              ; REMOVE BUGCHK PRIVILEGE
02F0    393 ;
02F0    394         NEXT_TEST_CASE  SFSNE11
```

```
                                                L 1
02FC    395 ;
02FC    396 ;++
02FC    397 ; ******************************************************
02FC    398 ; *
02FC    399 ; *  TEST CASE NAME:              SFSNE11
02FC    400 ; *
02FC    401 ; *  SYSTEM SERVICE:              SN'ERR
02FC    402 ; *
02FC    403 ; *  ARGUMENT UNDER TEST:         MSGBUF_SNE11
02FC    404 ; *
02FC    405 ; *  INPUT CONDITIONS:
02FC    406 ; *     MESSAGE BUFFER DESCRIPTOR IN NON-ACCESSIBLE PSECT.
02FC    407 ; *
02FC    408 ; *  EXPECTED RESULTS:
02FC    409 ; *     1) SYSTEM STATUS CODE:  ACCVIO
02FC    410 ; *     2) REGISTERS R2 THROUGH FP UNCHANGED
02FC    411 ; *
02FC    412 ; ******************************************************
02FC    413 ;--
02FC    414 ;
02FC    415         PRIV    ADD,BUGCHK              ; RE-ACQUIRE PRIVILEGE REMOVED BY SFSNE10
031C    416 ;
031C    417         TCEND
```

```
                                M 1
031D    418 TS1:
031D    419         TESTSERV        FAO,ERR,SATS,                                   -
031D    420                                                                         -
031D    421            <1,CTRSTR_FAO,                                               -
031D    422                          CTRSTR_FA010,BADPARAM,  -  ; SFFA010
031D    423                                                  >,                     -
031D    424                                                                         -
031D    425            <1,OUTLEN_FAO,                                               -
031D    426                                                  >,                     -
031D    427                                                                         -
031D    428            <1,OUTBUF_FAO,                                               -
031D    429                          OUTBUF_FA030,BUFFEROVF, -  ; SFFA030
031D    430                                                  >,                     -
031D    431                                                                         -
031D    432            <1,P1_FAO,                                       -
031D    433                                                  >,                     -
031D    434                                                                         -
031D    435            <1,P2_FAO,                                       -
031D    436                                                  >,                     -
031D    437
062A    438         TS_CLEANUP              ; CLEAN UP & RETURN TO TEST_SERV_EXEC
```

```
064A    439 TS2:
064A    440         TESTSERV        FAOL,ERR,SATS,                          -
064A    441                                                                 -
064A    442            <1,CTRSTR_FAL,                                       -
064A    443                         CTRSTR_FAL10,BADPARAM,  -  ; SFFAL10
064A    444                                                 >,              -
064A    445                                                                 -
064A    446            <1,OUTLEN_FAL,                                       -
064A    447                                                 >,              -
064A    448                                                                 -
064A    449            <1,OUTBUF_FAL,                                       -
064A    450                         OUTBUF_FAL30,BUFFEROVF, -  ; SFFAL30
064A    451                                                 >,              -
064A    452                                                                 -
064A    453            <1,PRMLST_FAL,                                       -
064A    454                                                 >,              -
064A    455
08BE    456         TS_CLEANUP              ; CLEAN UP & RETURN TO TEST_SERV_EXEC
```

```
08DE    457 TS3:
08DE    458         TESTSERV        SNDERR,ERR,SATS,                                -
08DE    459                                                                         -
08DE    460            <1,MSGBUF_SNE,                                               -
08DE    461                         MSGBUF_SNE10,NOPRIV,    -  ; SFSNE10
08DE    462                         MSGBUF_SNE11,ACCVIO,    -  ; SFSNE11
08DE    463                                                   >,                    -
08DE    464
09A9    465         TS_CLEANUP              ; CLEAN UP & RETURN TO TEST_SERV_EXEC
```

```
                                09C9   466          .SBTTL  EXECUTE & CLEANUP
                                09C9   467 EXECUTE:
                                09C9   468          TEST_SERV_EXEC                  ; EXECUTE ALL T. CASES IN ALL GROUPS
                                09E7   469 CLEANUP:
                         0138    30    09E7   470          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE END MSG
00000044'EF    01    1C    01    F0    09EA   471          INSV    #1,#STS$V_INHIB_MSG,#1,MOD_MSG_CODE
                                09F3   472                                          ; INHIBIT PRINTING
                                09F3   473          $EXIT_S MOD_MSG_CODE            ; EXIT TO OP SYS WITH MSG CODE
```

```
                                    D 2
0A00    475          .SBTTL  TC_CONTROL
0A00    476 ;++
0A00    477 ; FUNCTIONAL DESCRIPTION:
0A00    478 ;
0A00    479 ;                   THE TC_CONTROL SUBROUTINE IS CALLED BY THE TEST_SERV_EXEC
0A00    480 ; MACRO TO EXECUTE A GROUP OF TEST CASES. A GROUP IS DEFINED BY A TC_GROUP
0A00    481 ; MACRO. FOR EACH TC_GROUP MACRO, THERE IS A CORRESPONDING TESTSERV MACRO.
0A00    482 ; TESTSERV CONTAINS CODE TO EXECUTE SYSTEM SERVICES AND CHECK THE RETURNED
0A00    483 ; STATUS CODE VALUES. TESTSERV ARGUMENTS ARE CODED TO SPECIFY ALL THE SYSTEM
0A00    484 ; SERVICE ARGUMENT VALUES AND THE EXPECTED STATUS CODE FOR EACH TEST CASE
0A00    485 ; DEFINED BY A NEXT_TEST_CASE MACRO WITHIN THE GROUP. TC_CONTROL USES A
0A00    486 ; CO-ROUTINE INTERFACE TO ENTER THE CODE OF THE APPROPRIATE TESTSERV MACRO
0A00    487 ; IN VARIOUS PLACES. THE FIRST ENTRY OCCURS ONCE PER GROUP TO ALLOW TESTSERV
0A00    488 ; TO DO SOME INITIALIZATION. THEN TWO ENTRIES ARE MADE FOR EACH TEST CASE IN
0A00    489 ; THE GROUP. THE FIRST ALLOWS TESTSERV TO ISSUE THE SUBJECT SYSTEM SERVICE.
0A00    490 ; THE SECOND ENTRY FOR THE TEST CASE CAUSES TESTSERV TO CHECK THE RETURNED
0A00    491 ; STATUS CODE, PRINTING A FAILURE MESSAGE IF IT IS NOT THE EXPECTED CODE.
0A00    492 ; IF THERE ARE NO MORE TEST CASES IN THE CURRENT GROUP, TESTSERV (NOT TC_CONTROL)
0A00    493 ; RETURNS DIRECTLY TO TEST_SERV_EXEC (RSB ACTUALLY ISSUED IN TS_CLEANUP MACRO)
0A00    494 ; FROM THIS SECOND ENTRY; OTHERWISE, CONTROL RETURNS TO TC_CONTROL WHICH
0A00    495 ; IN TURN ENTERS TESTSERV AGAIN FOR THE NEXT TEST CASE. THE FAILURE OF A
0A00    496 ; TEST CASE DOES NOT CAUSE TERMINATION OF THE TEST MODULE.
0A00    497 ;
0A00    498 ; CALLING SEQUENCE:
0A00    499 ;
0A00    500 ;       BSBW TC_CONTROL (ISSUED WITHIN THE TEST_SERV_EXEC MACRO)
0A00    501 ;                       (RSB IS ISSUED WITHIN THE TS_CLEANUP MACRO)
0A00    502 ;
0A00    503 ; INPUT PARAMETERS:
0A00    504 ;
0A00    505 ;       NONE
0A00    506 ;
0A00    507 ; IMPLICIT INPUTS:
0A00    508 ;
0A00    509 ;       ARGUMENTS SPECIFIED ON EACH TESTSERV MACRO MAY BE VIEWED AS
0A00    510 ;       INPUTS, SINCE TC_CONTROL AND TESTSERV ACT AS CO-ROUTINES.
0A00    511 ;
0A00    512 ; OUTPUT PARAMETERS:
0A00    513 ;
0A00    514 ;       SEVERITY CODE FIELD OF MOD_MSG_CODE (BITS 0,1,2) IS SET TO ERROR
0A00    515 ;       IF ANY TEST CASE IN THE CURRENT GROUP FAILS; OTHERWISE IT REMAINS
0A00    516 ;       SET TO SUCCESSFUL.
0A00    517 ;
0A00    518 ; IMPLICIT OUTPUTS:
0A00    519 ;
0A00    520 ;       %UETP-I-TEXT,   ERROR MESSAGES ARE WRITTEN TO SYS$OUTPUT BY
0A00    521 ;                       THE TESTSERV MACRO (CO-ROUTINE WITH TC_CONTROL)
0A00    522 ;
0A00    523 ; COMPLETION CODES:
0A00    524 ;
0A00    525 ;       NONE
0A00    526 ;
0A00    527 ; SIDE EFFECTS:
0A00    528 ;
0A00    529 ;       NONE
0A00    530 ;
0A00    531 ;--
```

```
                                0A00    532
                                0A00    533
                                0A00    534
                                0A00    535 TC_CONTROL:
            00000064'EF   DD    0A00    536         PUSHL   TS_EP                      ; PUSH TESTSERV ENTRY POINT
                          9E    16  0A06    537         JSB     @(SP)+                  ; ENTER TESTSERV INITIALIZATION
                                0A08    538 10$:                                    ; PROCESS NEXT TEST CASE
        00000056'EF     20    90  0A08    539         MOVB    #^A/ /,$$TSTN$$+2       ; MAKE SURE T.C. NAME HAS A BLANK
                        002F  30  0A0F    540         BSBW    REG_SAVE                ; SAVE REGISTERS
            00000004'FF       16  0A12    541         JSB     @CURRENT_TC             ; JUMP TO CURRENT TEST CASE
                        0037  30  0A18    542         BSBW    REG_REST                ; RESTORE REGS FOR TESTSERV
                          9E    16  0A1B    543         JSB     @(SP)+                  ; LET TESTSERV ISSUE SYSTEM SERVICE
                        0042  30  0A1D    544         BSBW    REG_COMP                ; COMPARE REGS TO SEE IF ...
                                0A20    545                                         ; ... SYSTEM SERVICE CHANGED ANY
                          9E    16  0A20    546         JSB     @(SP)+                  ; LET TESTSERV CHEK S.S. STATUS CODE
        00000056'EF     2A    91  0A22    547         CMPB    #^A/*/,$$TSTN$$+2       ; HAS TESTSERV INDICATED FAILURE ?
                        DD    12  0A29    548         BNEQU   10$                     ; NO -- PROCESS NEXT TEST CASE
 00000060'EF    00000088'EF   DE    0A2B    549         MOVAL   TEST_MOD_FAIL,TMD_ADDR  ; YES -- INDICATE FAILED IN END MSG
00000044'EF   03    00    02    F0  0A36    550         INSV    #ERROR,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR ERROR
                        C7    11  0A3F    551         BRB     10$                     ; LOOP BAK TO PROCESS NEXT TEST CASE
                                0A41    552 ;
                                0A41    553 ;   TC_CONTROL RETURNS TO TEST_SERV_EXEC VIA TESTSERV (IN TS_CLEANUP MACRO)
                                0A41    554 ;
```

```
                                0A41    556          .SBTTL  SUBROUTINES
                                0A41    557 REG_SAVE:
                                0A41    558 ;
                                0A41    559 ;  ************************************************************************
                                0A41    560 ;  *                                                                      *
                                0A41    561 ;  *  SAVES R0 THRU SP IN REG_SAVE_AREA                                   *
                                0A41    562 ;  *                                                                      *
                                0A41    563 ;  ************************************************************************
                                0A41    564 ;
                    7FFF 8F     BB  0A41    565          PUSHR   #R0_THRU_SP             ; SAVE ALL REGS ON STACK
        00000008'EF 6E     3C   28  0A45    566          MOVC3   #60,(SP),REG_SAVE_AREA  ; SAVE REGS (BEFORE S.S.)
                    7FFF 8F     BA  0A4D    567          POPR    #R0_THRU_SP             ; CLEAN UP STACK
                                05  0A51    568          RSB                             ; .... AND RETURN
                                0A52    569 ;
                                0A52    570 ;
                                0A52    571 ;
                                0A52    572 ;
                                0A52    573 REG_REST:
                                0A52    574 ;
                                0A52    575 ;
                                0A52    576 ;  ************************************************************************
                                0A52    577 ;  *                                                                      *
                                0A52    578 ;  *  RESTORES R0 THRU SP FROM REG_SAVE_AREA                              *
                                0A52    579 ;  *                                                                      *
                                0A52    580 ;  ************************************************************************
                                0A52    581 ;
                    5E     3C   C2  0A52    582          SUBL2   #60,SP                  ; MOVE SP TO MAKE ROOM FOR REGS
        6E  00000008'EF    3C   28  0A55    583          MOVC3   #60,REG_SAVE_AREA,(SP)  ; MOVE REGS ONTO STACK FOR POP
                    7FFF 8F     BA  0A5D    584          POPR    #R0_THR0_SP             ; RESTORE ALL REGS FOR TESTSERV
                                05  0A61    585          RSB                             ; ... AND RETURN
```

```
                     0A62   587 REG_COMP:
                     0A62   588 ;
                     0A62   589 ;  ************************************************************************
                     0A62   590 ;  *                                                                      *
                     0A62   591 ;  *  1) PUSHES ALL REGS ONTO STACK                                        *
                     0A62   592 ;  *  2) COMPARES REGISTER IMAGES FROM STACK WITH CORRESPONDING             *
                     0A62   593 ;  *       IMAGES FROM REG_SAVE_AREA FOR ALL REGISTERS SPECIFIED           *
                     0A62   594 ;  *       IN REG_COMP_MASK.                                              *
                     0A62   595 ;  *  3) FOR EACH UNEQUAL COMPARE, AN ERROR MESSAGE IS PRINTED             *
                     0A62   596 ;  *       (USING $FAO AND $OUTPUT SYSTEM SERVICES).                      *
                     0A62   597 ;  *  4) POPS ALL REGS OFF OF STACK                                        *
                     0A62   598 ;  *                                                                      *
                     0A62   599 ;  ************************************************************************
                     0A62   600 ;
        7FFF 8F   BB 0A62   601         PUSHR   #^RO_THRU_SP                    ; SAVE ALL REGISTERS ON STACK
56 00000008'EF    DE 0A66   602         MOVAL   REG_SAVE_AREA,R6                ; POINT R6 TO BEG OF ...
                     0A6D   603                                                 ; ... REGS (BEFORE S.S.)
          54   5E   D0 0A6D   604         MOVL    SP,R4                           ; POINT R4 TO BEG OF
                     0A70   605                                                 ; ... REGS (AFTER S.S.)
          53   FF 8F   98 0A70   606         CVTBL   #-1,R3                          ; INITIALIZE REG_COMP_MASK INDEX
                     0A74   607 REG_COMP_NEXT:
                53   D6 0A74   608         INCL    R3                              ; POINT TO NEXT BIT IN MASK
                53   0F 91 0A76   609         CMPB    #15,R3                          ; END OF THE MASK ?
                     03 1A 0A79   610         BGTRU   REG_COMP_CONT                   ; NO -- CONTINUE
                   009F 31 0A7B   611         BRW     REG_COMP_RSB                    ; YES -- GO TO COMMON RETURN
                     0A7E   612 REG_COMP_CONT:
                84   86 D1 0A7E   613         CMPL    (R6)+,(R4)+                     ; REG BEFORE = REG AFTER ?
                     F1 13 0A81   614         BEQLU   REG_COMP_NEXT                   ; YES -- LOOK FOR NEXT REG
     E9 00000000'EF    53 E1 0A83   615         BBC     R3,REG_COMP_MASK,REG_COMP_NEXT
                     0A8B   616                                                 ; NO -- GET NEXT IF BIT NOT SET
        00000048'EF    53 D0 0A8B   617         MOVL    R3,CLOB_REG_NO                  ; NO -- GIVE REG NUMBER TO FAO
     0000004C'EF    FC A6 D0 0A92   618         MOVL    -4(R6),REG_BEFORE_SS            ; GIVE "BEFORE" CONTENTS TO FAO
     00000050'EF    FC A4 D0 0A9A   619         MOVL    -4(R4),REG_AFTER_SS             ; GIVE "AFTER" CONTENTS TO FAO
        00000056'EF    2A 90 0AA2   620         MOVB    #^A/*/,$$TSTN$$+2               ; GIVE FAILURE INDIC'N IN ERROR MSG
                     0AA9   621 ;
                     0AA9   622         $FAO_S  ERR_MSG_FAOCTL,OUTL,OUTD,$$SNAD$$,  -
                     0AA9   623                 $$ASEQ$$,$$PSEQ$$,CLOB_REG_NO,REG_BEFORE_SS,REG_AFTER_SS
                     0ADC   624 ;
     F631 CF    F5FB CF   B0 0ADC   625         MOVW    OUTL,OUTD                       ; ACTUAL OUTPUT LEN IN STRING DESC'R
                     0AE3   626         PUTMSG  <#UETP$_TEXT,#1,#OUTD>          ; PRINT THE MSG
     F615 CF    0084 8F   B0 0AF8   627         MOVW    #OUTE-OUTB,OUTD                 ; GET MAX LEN BACK INTO DESCRIPTOR
      00000056'EF    20 90 0AFF   628         MOVB    #^A/ /,$$TSTN$$+2               ; REMOVE FAIL INDIC'N FOR NEXT MSG
00000060'EF   00000088'EF   DE 0B06   629         MOVAL   TEST_MOD_FAIL,TMD_ADDR          ; INDICATE FAILED IN END MSG
00000044'EF   03   00   02  F0 0B11   630         INSV    #ERROR,#0,#3,MOD_MSG_CODE       ; ADJUST STATUS CODE FOR ERROR
                 FF57   31 0B1A   631         BRW     REG_COMP_NEXT                   ; GO LOOK FOR NEXT REG TO COMPARE
                     0B1D   632 REG_COMP_RSB:
                7FFF 8F   BA 0B1D   633         POPR    #^RO_THRU_SP                    ; CLEAN UP STACK
                     05 0B21   634         RSB                                     ; RETURN TO CALLER
```

```
                                H 2
0B22    636 MOD_MSG_PRINT:
0B22    637 ;
0B22    638 ;  ****************************************************************
0B22    639 ;  *                                                              *
0B22    640 ;  *  PRINTS THE TEST MODULE BEGUN/SUCCESSFUL/FAILED MESSAGES     *
0B22    641 ;  *      (USING THE PUTMSG MACRO).                               *
0B22    642 ;  *                                                              *
0B22    643 ;  ****************************************************************
0B22    644 ;
0B22    645         PUTMSG  <MOD_MSG_CODE,#2,TMN_ADDR,TMD_ADDR> ; PRINT MSG
05  0B3D    646         RSB                                     ; ... AND RETURN TO CALLER
0B3E    647 ;
0B3E    648 CHMRTN:
0B3E    649 ; ***************************************************************
0B3E    650 ; *
0B3E    651 ; *     CHANGE MODE ROUTINE. THIS ROUTINE GETS CONTROL WHENEVER
0B3E    652 ; *     A CMKRNL, CMEXEC, OR CMSUP SYSTEM SERVICE IS ISSUED
0B3E    653 ; *     BY THE MODE MACRO ('TO' OPTION).  IT MERELY DOES
0B3E    654 ; *     A JUMP INDIRECT ON A FIELD SET UP BY MODE. IT HAS
0B3E    655 ; *     THE EFFECT OF RETURNING TO THE END OF THE MODE
```

```
                     0B3E    657 ; *     MACRO EXPANSION.
                     0B3E    658 ; *
                     0B3E    659 ; ***************************************************************
                     0B3E    660 ;
                0000 0B3E    661         .WORD   0                       ; ENTRY MASK
00000079'FF       17 0B40    662         JMP     @CHM_CONT               ; RETURN TO MODE MACRO IN NEW MODE
                     0B46    663 ;
                     0B46    664 ; *     RET INSTR WILL BE ISSUED IN EXPANSION OF 'MODE FROM, ....' MACRO
                     0B46    665 ;
                     0B46    666         .END    SATSSF16
```

```
$$$CHARS                = 00000048
$$$FIRSTTC$$$           = 00000000
$$$STRINGS              = 00000000
$$ACT$$                   000000F3 R      06
$$ARG$$                   000000FB R      06
$$ASEQ$$                  000000EB R      06
$$CALL$$                  000000DF R      06
$$DISP$$                  000001E6 R      06
$$ERR$$                   000001A0 R      06
$$EXP$$                   000000F7 R      06
$$INIT$$                  000000E3 R      06
$$MAXP$$                = 00000005
$$PSEQ$$                  000000EF R      06
$$SNAD$$                  000000E7 R      06
$$T1                    = 00000004
$$T2                    = 00000009
$$TSTN$$                  00000054 R      03
CHMRTN                    00000B3E R      06
CHM_CONT                  00000079 R      03
CLEANUP                   000009E7 R      06
CLOB_REG_NO               00000048 R      03
COUNTED_STR               000000E3 R      02
CTL$GL_PHD                ******** X      06
CTRSTR_FAL                000000BD R      02
CTRSTR_FAL10              000000CE R      02
CTRSTR_FAO                000000BD R      02
CTRSTR_FAO10              000000CE R      02
CURRENT_TC                00000004 R      03
EMPTY                     00000000 R      04
ERROR                   = 00000002
ERR_MSG_FAOCTL            00000002 R      02
EXECUTE                   000009C9 R      06
GRP_TOTAL               = 00000003
INADR                     000000A9 R      02
INFO                    = 00000003
LIB$SIGNAL                ******** X      06
MEXIT                   = 00000000
MOD_MSG_CODE              00000044 R      03
MOD_MSG_PRINT             00000B22 R      06
MSGBUF_SNE                00000102 R      02
MSGBUF_SNE10              00000102 R      02
MSGBUF_SNE11              00000008 R      05
MY_DISR                   00000123 R      02
NARGS                   = 00000012
NOACCESS                  00000000 R      05
NSSARGS                 = 00000001
ONES                      000000B5 R      02
OUTB                      0000011C R      06
OUTBUF_FAL                00000093 R      03
OUTBUF_FAL30              000000CD R      03
OUTBUF_FAO                00000093 R      03
OUTBUF_FAO30              000000CD R      03
OUTD                      00000114 R      06
OUTE                      000001A0 R      06
OUTL                      000000DB R      06
OUTLEN_FAL                00000091 R      03
OUTLEN_FAO                00000091 R      03
P1_FAO                    000000DF R      02
P2_FAO                    000000FA R      02
PHD$Q_PRIVMSK           = 00000000
PRIVMASK                  00000071 R      03
PRIV_ARGS               = 00000002
PRMLST_FAL                000000FE R      02
PROT                      000000B1 R      02
PRT$C_NA                  ********    X   02
PRV$V_BUGCHK            = 00000017
PRVPRT                    00000070 R      03
RO_THRU_SP              = 00007FFF
REGS                      0000007D R      03
REG_AFTER_SS              00000050 R      03
REG_BEFORE_SS             0000004C R      03
REG_COMP                  00000A62 R      06
REG_COMP_CONT             00000A7E R      06
REG_COMP_MASK             00000000 R      02
REG_COMP_NEXT             00000A74 R      06
REG_COMP_RSB              00000B1D R      06
REG_REST                  00000A52 R      06
REG_SAVE                  00000A41 R      06
REG_SAVE_AREA             00000008 R      03
RETADR                    00000068 R      03
SATSSF16                  00000000 R      06
SEVERE                  = 00000004
SHR$K_SHRDEF            = 00000001
SHR$_TEXT               = 00001130
SS$_ACCVIO                ********    X   06
SS$_BADPARAM              ********    X   06
SS$_BUFFEROVF             ********    X   06
SS$_NOPRIV                ********    X   06
STS$V_INHIB_MSG         = 0000001C
SUCCESS                 = 00000001
SYS$CMKRNL                ********   GX   06
SYS$EXIT                  ********   GX   06
SYS$FAO                   ********    X   06
SYS$FAOL                  ********   GX   06
SYS$HIBER                 ********   GX   06
SYS$SETPRN                ********   GX   06
SYS$SETPRT                ********   GX   06
SYS$SETPRV                ********   GX   06
SYS$SNDERR                ********   GX   06
SYS$WAKE                  ********   GX   06
TC1                       00000241 R      06
TC2                       00000275 R      06
TC3                       000002A9 R      06
TCG_NO                  = 00000003
TC_CONTROL                00000A00 R      06
TEST_MOD_BEG              00000077 R      02
TEST_MOD_FAIL             00000088 R      02
TEST_MOD_NAME             0000006E R      02
TEST_MOD_NAME_D           0000008F R      02
TEST_MOD_SUCC             0000007D R      02
TMD_ADDR                  000000B0 R      03
TMN_ADDR                  0000005C R      03
TPID                      00000000 R      03
TS1                       0000031D R      06
```

```
TS2                             0000064A R      06
TS3                             000008DE R      06
TS_EP                           00000064 R      03
TTNAME                          0000009F R      02
UETP$_SATSMS                  = 007480D9
UETP$_TEXT                    = 00741133
WARNING                       = 00000000
```

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	00000133 (307.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	000000D6 (214.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC BYTE
SATS_ACCVIO_1	00000200 (512.)	04 (4.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC PAGE
SATS_ACCVIO_2	00000200 (512.)	05 (5.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC PAGE
SATSSF16	00000846 (2886.)	06 (6.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.06	00:00:00.30
Command processing	112	00:00:00.65	00:00:01.84
Pass 1	395	00:00:15.21	00:00:23.43
Symbol table sort	0	00:00:01.06	00:00:01.09
Pass 2	141	00:00:03.45	00:00:04.32
Symbol table output	16	00:00:00.09	00:00:00.11
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	699	00:00:20.55	00:00:31.13

The working set limit was 1500 pages.
80713 bytes (158 pages) of virtual memory were used to buffer the intermediate code.
There were 40 pages of symbol table space allocated to hold 601 non-local and 109 local symbols.
666 source lines were read in Pass 1, producing 28 object records in Pass 2.
61 pages of virtual memory were used to define 45 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SHRLIB]UETP.MLB;1	19
_255DUA28:[SYS.OBJ]LIB.MLB;1	2
_255DUA28:[SYSLIB]STARLET.MLB;2	18
TOTALS (all libraries)	39

1216 GETS were required to define 39 macros.

There were no errors, warnings or information messages.

```
MACRO/LIS=LIS$:SATSSF16/OBJ=OBJ$:SATSSF16 MSRC$:SATSSF16/UPDATE=(ENH$:SATSSF16)+EXECML$/LIB+SHRLIB$:UETP/LIB
```

SATSSF15
LIS

SATSSF16
LIS

SATSSF14
LIS

SATSSF12
LIS

SATSSF13
LIS

SATSSS08
LIS

SATSSS05
LIS

SATSSS22
LIS

SATSSS07
LIS

SATSSS01
LIS