F 13

```
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPrPPPPPPPPP        SSSSSSSSSSSS  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
UUU         UUU  EEE                    TTT        PPP                          SSS        YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSSS           YYY
```

```
_$

Va
--
00
00
00
00
00
48
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
```

```
**FILE**ID**SUCCOMMON                                          D  5



            SSSSSSSS  UU      UU    CCCCCCCC    CCCCCCCC    OOOOOO    MM      MM  MM      MM    OOOOOO    NN      NN
            SSSSSSSS  UU      UU    CCCCCCCC    CCCCCCCC    OOOOOO    MM      MM  MM      MM    OOOOOO    NN      NN
          SS          UU      UU  CC          CC          OO      OO  MMMM  MMMM  MMMM  MMMM  OO      OO  NN      NN
          SS          UU      UU  CC          CC          OO      OO  MMMM  MMMM  MMMM  MMMM  OO      OO  NN      NN
          SS          UU      UU  CC          CC          OO      OO  MM  MM  MM  MM  MM  MM  OO      OO  NNNN    NN
          SS          UU      UU  CC          CC          OO      OO  MM  MM  MM  MM  MM  MM  OO      OO  NNNN    NN
            SSSSSS    UU      UU  CC          CC          OO      OO  MM      MM  MM      MM  OO      OO  NN  NN  NN
            SSSSSS    UU      UU  CC          CC          OO      OO  MM      MM  MM      MM  OO      OO  NN  NN  NN
                  SS  UU      UU  CC          CC          OO      OO  MM      MM  MM      MM  OO      OO  NN    NNNN
                  SS  UU      UU  CC          CC          OO      OO  MM      MM  MM      MM  OO      OO  NN    NNNN
                  SS  UU      UU  CC          CC          OO      OO  MM      MM  MM      MM  OO      OO  NN      NN      ....
                  SS  UU      UU  CC          CC          OO      OO  MM      MM  MM      MM  OO      OO  NN      NN      ....
          SSSSSSSS    UUUUUUUUUU    CCCCCCCC    CCCCCCCC    OOOOOO    MM      MM  MM      MM    OOOOOO    NN      NN      ....
          SSSSSSSS    UUUUUUUUUU    CCCCCCCC    CCCCCCCC    OOOOOO    MM      MM  MM      MM    OOOOOO    NN      NN      ....

          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
```

SU(
V0

00(

Table of contents

SU
VO

```
0000            .TITLE  SUCCOMMON SATS SYSTEM SERVICE TESTS   COMMON (SUCC S.C.)
0000            .IDENT  'V04-000'
0000
0000     ;
0000     ;*****************************************************************************
0000     ;*                                                                           *
0000     ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     ;*  ALL RIGHTS RESERVED.                                                     *
0000     ;*                                                                           *
0000     ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000     ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
0000     ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000     ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000     ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000     ;*  TRANSFERRED.                                                             *
0000     ;*                                                                           *
0000     ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000     ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0000     ;*  CORPORATION.                                                             *
0000     ;*                                                                           *
0000     ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000     ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000     ;*                                                                           *
0000     ;*                                                                           *
0000     ;*****************************************************************************
0000     ;
0000
0000     ;++
0000     ; FACILITY:     SYSTST (SATS SYSTEM SERVICE TESTS)
0000     ;
0000     ; ABSTRACT:
0000     ;
0000     ;               THE SUCCOMMON MODULE CONTAINS SUBROUTINES WHICH, WHEN
0000     ; LINKED WITH EACH OF THE OBJECT MODULES SATSSS  .OBJ, FORM THE TEST
0000     ; MODULES SAT SS  .EXE TO TEST SUCCESSFUL OPERATION OF THE VMS SYSTEM
0000     ; SERVICES. THE SUCCOMMON MODULE CONTAINS SEVERAL COMMON SUBROUTINES
0000     ; USED BY EACH OF THE SATSSS  MODULES PLUS THE SUCCOMMON ROUTINE,
0000     ; WHICH DISTRIBUTES CONTROL TO ALL SUBROUTINES.
0000     ;
0000     ; ENVIRONMENT:  USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000     ;               DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000     ;
0000     ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: APR, 1977
0000     ;
0000     ; MODIFIED BY:
0000     ;
0000     ;       , : VERSION
0000     ; 01    -
0000     ;--
```

```
                 0000          .SBTTL  DECLARATIONS
                 0000  ;
                 0000  ; INCLUDE FILES:
                 0000  ;
                 0000          $STSDEF                             ; DEFINE MSG CODE SYMBOLS
                 0000          $UETPDEF                            ; GENERATE UETP MESSAGE DEFS
                 0000          $SHR_MESSAGES UETP,116,<<TEXT,INFO>> ; DEFINE UETP$_TEXT MSG
                 0000  ;
                 0000  ; MACROS:
                 0000  ;
                 0000  ;
                 0000  ; EQUATED SYMBOLS:
                 0000  ;
00000000         0000  WARNING         = 0                         ; WARNING SEVERITY VALUE
00000001         0000  SUCCESS         == 1                        ; SUCCESS SEVERITY VALUE
00000002         0000  ERROR           = 2                         ; ERROR SEVERITY VALUE
00000003         0000  INFO            = 3                         ; INFORMATIONAL SEVERITY VALUE
00000004         0000  SE.ERE          = 4                         ; SEVERE (FATAL) SEVERITY VALUE
00000082         0000  BUFFL           = 130                       ; LENGTH OF FAO OUTPUT BUFFER
0000000C         0000  CMPC_SAV        == ^M<R2,R3>                ; REGISTER SAVE MASK FOR CMPC INSTRUCTION
00007FFF         0000  R0_THRU_SP      == ^M<R0,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,AP,FP,SP>
                 0000  ;
                 0000  ; OWN STORAGE:
                 0000  ;
```

```
                   00000000     77           .PSECT  ROD_COMM,RD,NOWRT,NOEXE,LONG
FFFFFFFF FFFFFFFF  0000     78 ONES::             .LONG   -1,-1              ; A QUADWORD OF 1 BITS
                   0008     79 TEST_MOD_BEG:      STRING  C,<begun>          ; DISPOSITION FIELD OF TEST MOD MSG
                   000E     80 TEST_MOD_SUCC::    STRING  C,<successful>     ; DISPOSITION FIELD OF TEST MOD MSG
                   0019     81 TEST_MOD_FAIL:     STRING  C,<failed>         ; DISPOSITION FIELD OF TEST MOD MSG
                   0020     82 MSG2_CND_CTL1:     STRING  I,<!15* !AC: !AC> ; FIRST FORM OF MSG2
                   0035     83 MSG2_CND_CTL2:     STRING  I,<!15* !AC ARGUMENT; !AC: !AS>
                   0058     84                                               ; SECOND FORM OF MSG2
                   0058     85 MSG2_CND_CTL3:     STRING  I,<!15* !AC ARGUMENT; !AC: !#(#XL)>
                   007F     86                                               ; THIRD FORM OF MSG2
                   007F     87                                               ; NOTE MSG3_ERR_CTL IS IN A DIFF PSECT
                   007F     88 MSG4_ERR_CTL1:     STRING  I,<!15* !AC VALUE: !AS>
                   009A     89                                               ; FIRST FORM FOR MSG4
                   009A     90 MSG4_ERR_CTL2:     STRING  I,<!15* !AC VALUE: !#(#XL)>
                   00B9     91                                               ; 2ND FORM FOR MSG4
                   00B9     92 EXP:               STRING  C,<EXPECTED>       ; COUNTED STRING FOR MSG4
                   00C2     93 REC:               STRING  C,<RECEIVED>       ; COUNTED STRING FOR MSG4
                   00CB     94 APC:               STRING  C,<APPROXIMATE PC> ; COUNTED STRING FOR MSG 4
```

```
         00000000                .PSECT  RWD_COMM,RD,WRT,NOEXE,LONG
0000003C     0000    REG_SAVE_AREA:  .BLKL    15                ; SAVE AREA FOR ALL REGS (SANS PC)
    0000     003C    TESTNUM::       .WORD    0                 ; SEQUENTIAL TEST NUMBER
00000042     003E    CHM_CONT::      .BLKL    1                 ; CHANGE MODE CONTINUE ADDRESS
007480D9     0042    MOD_MSG_CODE::  .LONG    UETP$_SATSMS      ; TEST MODULE MSG CODE FOR PUTMSG
00000000'    0046    TMN_ADDR:       .ADDRESS TEST_MOD_NAME     ; ADDR OF TEST MOD NAME FOR FAO
00000008'    004A    TMD_ADDR::      .ADDRESS TEST_MOD_BEG      ; ADDR OF T.M. DISP FIELD FOR FAO
00000052     004E    FAO_LEN::       .BLKL    1                 ; ACTUAL LENGTH RETURNED BY FAO
             0052    FAO_DESC::      STRING   0,BUFFL           ; BUFFER INTO WHICH FAO CREATES STRING
000000DD     00DC    MSG_CTXT::      .BLKB    1                 ; CONTEXT VALUE FOR ITEMS IN MESSAGES
000000E1     00DD    MSG_DATA1::     .BLKL    1                 ; INPUT PARAMETER FOR VARIOUS MSG WRITE RTNS
000000E5     00E1    MSG_A::         .BLKL    1                 ; LONGWORD PARAMETER USED BY FAO
000000E9     00E5    MSG_B::         .BLKL    1                 ; LONGWORD PARAMETER USED BY FAO
000000ED     00E9    MSG_C:          .BLKL    1                 ; LONGWORD PARAMETER USED BY FAO
000000F1     00ED    MSG_D:          .BLKL    1                 ; LONGWORD PARAMETER USED BY FAO
             00F1    ;
             00F1    ; NOTE -- FOLLOWING TWO PARAMETERS ARE INITIALIZED TO ENSURE
             00F1    ;         ACCESSIBILITY WHEN FAO SYSTEM SERVICE REFERENCES
             00F1    ;         THEIR CONTENTS.
             00F1    ;
000000F1'    00F1    MSG_E:          .ADDRESS MSG_E             ; LONGWORD PARAMETER USED BY FAO
000000F5'    00F5    MSG_F:          .ADDRESS MSG_F             ; LONGWORD PARAMETER USED BY FAO
00000101     00F9    EXPV::          .BLKQ    1                 ; EXPECTED VALUE FOR VERIFY ROUTINE
00000109     0101    RECV::          .BLKQ    1                 ; RECEIVED VALUE FOR VERIFY ROUTINE
0000010D     0109    PCV::           .BLKL    1                 ; SAVE AREA FOR PC VALUE
      00     010D    CFLAG::         .BYTE    0                 ; CONDITIONS FLAG: ZERO MEANS NO CONDS MSG
      00     010E    EFLAG::         .BYTE    0                 ; ERROR FLAG: ZERO MEANS NO ERRORS YET
00000110     010F    CONFLICT::      .BLKB    1                 ; CONFLICT INDICATOR ; 0 MEANS ...
             0110                                               ; ... NO CONDITION TABLE CONFLICT
         00000000                .PSECT  SUCCOMMON,RD,WRT,EXE
```

```
0000                  .SBTTL  SUCCOMMON
0000    ;++
0000    ; FUNCTIONAL DESCRIPTION:
0000    ;
0000    ;                    THE SUCCOMMON ROUTINE IS ENTERED WHENEVER A SATS TEST
0000    ; MODULE IS RUN TO TEST ONE OR MORE SYSTEM SERVICE(S), EXPECTING SUCCESS
0000    ; STATUS CODES. THE BASIC FUNCTION OF THE ROUTINE IS TO CAUSE MULTIPLE
0000    ; INVOCATIONS OF THE VERIFY SUBROUTINE, WHICH ISSUES AND VERIFIES THE
0000    ; SUBJECT SYSTEM SERVICE(S). FOLLOWING THE INVOCATIONS, THE ROUTINE EXITS
0000    ; TO THE OPERATING SYSTEM WITH A STATUS CODE INDICATING SUCCESS OR FAILURE
0000    ; OF THE TEST MODULE. VERIFY IS CALLED ONCE FOR EACH COMBINATION OF
0000    ; VALUES IN THE CONDITION TABLES. THERE MAY BE FROM ONE TO FIVE CONDITION
0000    ; TABLES, WITH FROM ONE TO FIVE ENTRIES EACH. THEREFORE VERIFY MAY BE
0000    ; CALLED FROM 1 TO 3125 TIMES, DEPENDING ON THE QUANTITY AND SIZES OF
0000    ; THE CONDITION TABLES. EACH TABLE REPRESENTS A VARIABLE IN SYSTEM
0000    ; SERVICE SPECIFICATION OR SYSTEM ENVIRONMENT; A TYPICAL EXAMPLE OF
0000    ; A VARIABLE IS ONE OF THE SYSTEM SERVICE INPUT ARGUMENTS. EACH TABLE
0000    ; ENTRY (ELEMFNT) REPRESENTS A VALUE FOR THE VARIABLE REPRESENTED BY
0000    ; THE TABLE. THE CURRENTLY SELECTED ELEMENT FOR EACH TABLE IS REPRESENTED
0000    ; BY A VALUE (FROM 1 TO 5) IN AN INDEX REGISTER (R2-R6 FOR CONDITION
0000    ; TABLES 1-5, RESPECTIVELY). SUCCOMMON CONTAINS A CALL TO THE VERIFY
0000    ; SUBROUTINE AT THE CENTER OF A GROUP OF NESTED LOOPS, ONE LOOP FOR EACH
0000    ; TABLE. THEREFORE, UPON EACH ENTRY TO VERIFY, REGISTERS R2-R6 CONTAIN
0000    ; VALUES REPRESENTING ONE COMBINATION OF CONDITION TABLE VALUES. AS A
0000    ; NEW ENTRY IN EACH TABLE IS SELECTED, A CONDITION TABLE SETUP SUBROUTINE
0000    ; IS CALLED (COND1,2,3,4,5) TO ALLOW TABLE-SPECIFIC SETUP CODE TO BE
0000    ; EXECUTED; ALSO, BEFORE ENTERING THE LOOP, SUCCOMMON CALLS A SETUP
0000    ; ROUTINE (TM_SETUP) TO EXECUTE TEST-MODULE-SPECIFIC SETUP CODE,
0000    ; APPLICABLE ACROSS ALL CONDITION TABLES. EACH OF THE SETUP SUBROUTINES
0000    ; AND THE VERIFY SUBROUTINE HAVE ASSOCIATED CLEANUP SUBROUTINES,
0000    ; EACH OF WHICH IS EXECUTED FOLLOWING VERIFY, ONCE FOR EACH EXECUTION
0000    ; OF ITS CORRESPONDING SETUP ROUTINE. THIS STRUCTURE GUARANTEES THAT
0000    ; CLEANUP (OF ACQUIRED RESOURCES, FOR EXAMPLE) IS ALWAYS PERFORMED FOR
0000    ; EACH SETUP, EVEN IN THE CASE WHERE AN ERROR PREMATURELY TERMINATES
0000    ; EXECUTION OF THE TEST MODULE. THE FIRST CALL TO VERIFY WHICH FINDS
0000    ; AN ERROR CAUSES TERMINATION OF THE TEST MODULE WITH A "TEST MODULE
0000    ; FAILED" MESSAGE.
0000    ;
0000    ; CALLING SEQUENCE:
0000    ;
0000    ;       $ RUN SATSSS__  ...  (DCL COMMAND)
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    ;       MESSAGES TO SYS$OUTPUT ARE THE ONLY OUTPUT FROM SUCCOMMON.
```

```
                      0000  ;         THEY ARE OF THE FORM:
                      0000  ;
                      0000  ;                 %UETP-S-SATSMS, TEST MODULE SATSSS__ BEGUN ... (BEGIN MSG)
                      0000  ;                 %UETP-S-SATSMS, TEST MODULE SATSSS__ SUCCESSFUL ... (END MSG)
                      0000  ;                 %UETP-E-SATSMS, TEST MODULE SATSSS__ FAILED ... (END MSG)
                      0000  ;                 %UETP-I-TEXT, ... (VARIABLE INFORMATION ABOUT A TEST MODULE FAILURE)
                      0000  ;
                      0000  ; COMPLETION CODES:
                      0000  ;
                      0000  ;         THE SUCCOMMON ROUTINE TERMINATES WITH A $EXIT TO THE
                      0000  ;         OPERATING SYSTEM WITH A STATUS CODE DEFINED BY UETP$_SATSMS.
                      0000  ;
                      0000  ; SIDE EFFECTS:
                      0000  ;
                      0000  ;         NONE
                      0000  ;
                      0000  ;--
                      0000
                      0000
                      0000
                      0000  SUCCOMMON:
                OFFC  0000          .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11>
                      0002                                                  ; ENTRY MASK
         FFFB'   30   0002          BSBW    TM_SETUP                        ; PERFORM TEST MODULE SETUP
     0000010E'EF 95   0005          TSTB    EFLAG                           ; IS ERROR BEING PROCESSED ?
           03    13   000B          BEQL    10$                             ; NO -- CONTINUE
         00E0    31   000D          BRW     TM_CU                           ; YES -- PERFORM TEST MODULE CLEANUP
                      0010  10$:
           52    D4   0010          CLRL    R2                              ; POINT TO FIRST ELEMENT IN CONDITION 1 TAB
                      0012  COND1_LOOP:
     0000010F'EF 94   0012          CLRB    CONFLICT                        ; INDICATE NO CONDITION TABLE CONFLICT
         FFE5'   30   0018          BSBW    COND1                           ; EXECUTE CONDITION 1 ROUTINE
     0000010E'EF 95   001B          TSTB    EFLAG                           ; IS AN ERROR BEING PROCESSED ?
           03    13   0021          BEQL    10$                             ; NO -- CONTINUE
         00BD    31   0023          BRW     COND1_CU                        ; YES -- PERFORM CONDITION 1 CLEANUP
                      0026  10$:
     0000010F'EF 95   0026          TSTB    CONFLICT                        ; WAS THERE A CONDITION TABLE CONFLICT ?
           03    13   002C          BEQL    20$                             ; NO -- CONTINUE
         00B2    31   002E          BRW     COND1_CU                        ; YES -- SKIP THIS ELEMENT IN CONDIT 1 TABLE
                      0031  20$:
           53    D4   0031          CLRL    R3                              ; POINT TO FIRST ELEMENT IN CONDITION 2 TAB
                      0033  COND2_LOOP:
     0000010F'EF 94   0033          CLRB    CONFLICT                        ; INDICATE NO CONDITION TABLE CONFLICT
         FFC4'   30   0039          BSBW    COND2                           ; EXECUTE CONDITION 2 ROUTINE
     0000010E'EF 95   003C          TSTB    EFLAG                           ; IS AN ERROR BEING PROCESSED ?
           03    13   0042          BEQL    10$                             ; NO -- CONTINUE
         008F    31   0044          BRW     COND2_CU                        ; YES -- GO PERFORM CONDITION 2 CLEANUP
                      0047  10$:
     0000010F'EF 95   0047          TSTB    CONFLICT                        ; WAS THERE A CONDITION TABLE CONFLICT ?
           03    13   004D          BEQL    20$                             ; NO -- CONTINUE
         0084    31   004F          BRW     COND2_CU                        ; YES -- SKIP THIS ELEMENT IN CONDIT 2 TABLE
                      0052  20$:
           54    D4   0052          CLRL    R4                              ; POINT TO FIRST ELEMENT IN CONDITION 3 TAB
                      0054  COND3_LOOP:
     0000010F'EF 94   0054          CLRB    CONFLICT                        ; INDICATE NO CONDITION TABLE CONFLICT
         FFA3'   30   005A          BSBW    COND3                           ; EXECUTE CONDITION 3 ROUTINE
     0000010E'EF 95   005D          TSTB    EFLAG                           ; IS AN ERROR BEING PROCESSED ?
```

```
                         64   12  0063                BNEQ    COND3_CU                  ; YES -- GO PERFORM CONDITION 3 CLEANUP
               0000010F'EF    95  0065                TSTB    CONFLICT                  ; WAS THERE A CONDITION  ABLE CONFLICT ?
                         5C   12  006B                BNEQ    COND3_CU                  ; YES -- SKIP THIS ELEMENT IN CONDIT 3 TABLE
                         55   D4  006D                CLRL    R5                        ; POINT TO FIRST ELEMENT IN CONDITION 4 TAB
                                  006F     COND4_LOOP:
               0000010F'EF    94  006F                CLRB    CONFLICT                  ; INDICATE NO CONDITION TABLE CONFLICT
                       FF88'  30  0075                BSBW    COND4                     ; EXECUTE CONDITION 4 ROUTINE
               0000010E'EF    95  0078                TSTB    EFLAG                     ; IS AN ERROR BEING PROCESSED ?
                         3C   12  007E                BNEQ    COND4_CU                  ; YES -- GO PERFORM CONDITION 4 CLEANUP
               0000010F'EF    95  0080                TSTB    CONFLICT                  ; WAS THERE A CONDITION TABLE CONFLICT ?
                         34   12  0086                BNEQ    COND4_CU                  ; YES -- SKIP THIS ELEMENT IN CONDIT 4 TABLE
                         56   D4  0088                CLRL    R6                        ; POINT TO FIRST ELEMENT IN CONDITION 5 TAB
                                  008A     COND5_LOOP:
               0000010F'EF    94  008A                CLRB    CONFLICT                  ; INDICATE NO CONDITION TABLE CONFLICT
                       FF6D'  30  0090                BSBW    COND5                     ; EXECUTE CONDITION 5 ROUTINE
               0000010E'EF    95  0093                TSTB    EFLAG                     ; IS AN ERROR BEING PROCESSED ?
                         14   12  0099                BNEQ    COND5_CU                  ; YES -- GO PERFORM CONDITION 5 CLEANUP
               0000010F'EF    95  009B                TSTB    CONFLICT                  ; WAS THERE A CONDITION TABLE CONFLICT ?
                         0C   12  00A1                BNEQ    COND5_CU                  ; YES -- SKIP THIS ELEMENT IN CONDIT 5 TABLE
                       FF5A'  30  00A3                BSBW    VERIFY                    ; ISSUE SYSTEM SERVICE AND VERIFY IT
                       FF57'  30  00A6                BSBW    VFY_CLEANUP               ; PERFORM VERIFY CLEANUP
               0000003C'EF    B6  00A9                INCW    TESTNUM                   ; GET NEXT TEST CASE NUMBER
                                  00AF     COND5_CU:
                       FF4E'  30  00AF                BSBW    COND5_CLEANUP             ; PERFORM CONDITION 5 CLEANUP
   FFCE 56    01   00000000'EF    9D  00B2                ACBB    COND5_H,#1,R6,COND5_LOOP  ; LOOP THRU CONDITION 5 TABLE
                                  00BC     COND4_CU:
                       FF41'  30  00BC                BSBW    COND4_CLEANUP             ; PERFORM CONDITION 4 CLEANUP
   FFA6 55    01   00000000'EF    9D  00BF                ACBB    COND4_H,#1,R5,COND4_LOOP  ; LOOP THRU CONDITION 4 TABLE
                                  00C9     COND3_CU:
                       FF34'  30  00C9                BSBW    COND3_CLEANUP             ; PERFORM CONDITION 3 CLEANUP
   FF7E 54    01   00000000'EF    9D  00CC                ACBB    COND3_H,#1,R4,COND3_LOOP  ; LOOP THRU CONDITION 3 TABLE
                                  00D6     COND2_CU:
                       FF27'  30  00D6                BSBW    COND2_CLEANUP             ; PERFORM CONDITION 2 CLEANUP
   FF50 53    01   00000000'EF    9D  00D9                ACBB    COND2_H,#1,R3,COND2_LOOP  ; LOOP THRU CONDITION 2 TABLE
                                  00E3     COND1_CU:
                       FF1A'  30  00E3                BSBW    COND1_CLEANUP             ; PERFORM CONDITION 1 CLEANUP
   FF22 52    01   00000000'EF    9D  00E6                ACBB    COND1_H,#1,R2,COND1_LOOP  ; LOOP THRU CONDITION 1 TABLE
                                  00F0     TM_CU:
                       FF0D'  30  00F0                BSBW    TM_CLEANUP                ; PERFORM TEST MODULE CLEANUP
  00000042'EF    01    1C    01   F0  00F3                INSV    #1,#STS$V_INHIB_MSG,#1,MOD_MSG_CODE
                                  00FC                                              ; INHIBIT PRINTING
                                  00FC                $EXIT_S MOD_MSG_CODE              ; EXIT TO OPERATING SYSTEM WITH MSG CODE
```

```
                                         0109          .SBTTL  SUBROUTINES
                                         0109  WRITE_MSG2::
                                         0109  ;
                                         0109  ;++
                                         0109  ; ***********************************************************
                                         0109  ; *
                                         0109  ; *     WRITE_MSG2 SUBROUTINE.  THIS ROUTINE FORMATS (USING FAO)
                                         0109  ; *     AND PRINTS (USING PUTMSG MACRO) INFORMATION ABOUT THE
                                         0109  ; *     CURRENT ELEMENT FOR A PARTICULAR CONDITION TABLE.
                                         0109  ; *
                                         0109  ; *     INPUTS:
                                         0109  ; *             MSG_A:          A LONGWORD FIELD CONTAINING
                                         0109  ; *                             THE ADDRESS OF A COUNTED
                                         0109  ; *                             STRING FOR THE TITLE OF THIS
                                         0109  ; *                             CONDITION.
                                         0109  ; *
                                         0109  ; *             MSG_B:          A LONGWORD FIELD CONTAINING
                                         0109  ; *                             THE ADDRESS OF A COUNTED
                                         0109  ; *                             STRING FOR THE CURRENT TEXT
                                         0109  ; *                             ELEMENT FOR THIS CONDITION.
                                         0109  ; *
                                         0109  ; *             MSG_CTXT:       A BYTE FIELD CONTAINING THE
                                         0109  ; *                             CONTEXT FOR THIS CONDITION,
                                         0109  ; *                             AS DEFINED IN ITS CONDITION
                                         0109  ; *                             TABLE.
                                         0109  ; *
                                         0109  ; *             MSG_DATA1:      A LONGWORD FIELD CONTAINING
                                         0109  ; *                             THE ADDRESS OF THE CURRENT
                                         0109  ; *                             DATA ELEMENT FOR THIS CONDITION
                                         0109  ; *                             (ONLY FOR CONDITIONS WHICH
                                         0109  ; *                             ARE SYSTEM SERVICE ARGUMENTS).
                                         0109  ; *
                                         0109  ; *     OUTPUTS:
                                         0109  ; *             CONDITION MESSAGE IS PRINTED.
                                         0109  ; *
                                         0109  ; ***********************************************************
                                         0109  ;--
                                         0109  ;
        00'8F    000000DC'EF     91      0109          CMPB    MSG_CTXT,#NOTARG        ; IS THIS CONDITION A S.S. ARGUMENT ?
                          28     12      0111          BNEQU   10$                     ; YES -- CONTINUE
                                         0113          $FAO_S  MSG2_CND_CTL1,FAO_LEN,FAO_DESC,MSG_A,MSG_B
                                         0138                                          ; NO -- SIMPLY FORMAT TEXT
                        00CE     31      0138          BRW     WRITE_MSG2X             ; ... WRITE MSG AND EXIT
                                         013B  10$:
        00'8F    000000DC'EF     91      013B          CMPB    MSG_CTXT,#DESC          ; IS THIS A STRING DESCRIPTOR ARGUMENT ?
                          39     12      0143          BNEQU   20$                     ; NO -- CONTINUE
  000000E9'EF    000000DD'EF     D0      0145          MOVL    MSG_DATA1,MSG_C         ; YES -- GIVE DESCRIPTOR ADDRESS TO FAO
                                         0150          $FAO_S  MSG2_CND_CTL2,FAO_LEN,FAO_DESC,MSG_A,MSG_B,MSG_C
                                         017B                                          ; FORMAT STRING ARGUMENT
                        008B     31      017B          BRW     WRITE_MSG2X             ; ... WRITE MSG AND EXIT
                                         017E  20$:
  000000F1'EF    000000DD'EF     D0      017E          MOVL    MSG_DATA1,MSG_E         ; GET DATA ADDRESS READY FOR FAO
        00'8F    000000DC'EF     91      0189          CMPB    MSG_CTXT,#QUAD          ; DOES THIS ARGUMENT HAVE QUAD CONTEXT ?
                          1C     12      0191          BNEQU   30$                     ; NO -- GO PROCESS ALL OTHER CONTEXTS
             000000E9'EF  02     9A      0193          MOVZBL  #2,MSG_C                ; YES -- INDICATE 2 DATA ITEMS TO FORMAT
             000000ED'EF  09     9A      019A          MOVZBL  #9,MSG_D                ; ... EACH 8 BYTES LONG + A BLANK
000000F5'EF  000000F1'EF  04     C1      01A1          ADDL3   #4,MSG_E,MSG_F          ; COMPUTE ADDR OF 2ND ITEM FOR FAO
```

```
                              1D   11  01AD              BRB      40$                      ; ... AND GO FORMAT MSG
                                       01AF      30$:
               000000E9'EF    01   9A  01AF              MOVZBL   #1,MSG_C                 ; INDICATE JUST 1 DATA ITEM TO FORMAT
      000000ED'EF   000000DC'EF    9A  01B6              MOVZBL   MSG_CTXT,MSG_D           ; GET CONTEXT READY FOR FAO
      000000ED'EF   000000ED'EF    C0  01C1              ADDL2    MSG_D,MSG_D              ; ... DOUBLE IT FOR LENGTH OF PRINTABLE DATA
                                       01CC      40$:
                                       01CC              $FAO_S   MSG2_CND_CTL3,FAO_LEN,FAO_DESC,MSG_A,   -
                                       01CC                       MSG_B,MSG_C,MSG_D,@MSG_E,@MSG_F
                                       0209                                                ; FORMAT NON-STRING ARGUMENT MSG
                                       0209      WRITE_MSG2X:
                            01F1   30  0209              BSBW     OUTPUT_MSG               ; WRITE FORMATTED CONDITION MSG
                                   05  020C              RSB                               ; ... AND RETURN TO CALLER
```

```
              020D  WRITE_MSG3:
              020D  ;
              020D  ;++
              020D  ; ********************************************************
              020D  ; *
              020D  ; *       WRITE_MSG3 SUBROUTINE.  THIS ROUTINE FORMATS AND
              020D  ; *       PRINTS A GENERAL ERROR MSG.  IT IS CALLED BY CODE
              020D  ; *       GENERATED IN THE EXPANSION OF THE ERR_EXIT MACRO.
              020D  ; *
              020D  ; *       INPUTS:
              020D  ; *               TESTNUM:        A WORD FIELD CONTAINING THE
              020D  ; *                               CURRENT TEST CASE NUMBER.
              020D  ; *               MSG_A           A LONGWORD FIELD CONTAINING
              020D  ; *                               THE ADDRESS OF A QUADWORD
              020D  ; *                               STRING DESCRIPTOR WHICH HAS
              020D  ; *                               TEXT DESCRIBING THE CURRENT
              020D  ; *                               ERROR.
              020D  ; *
              020D  ; *       OUTPUTS:
              020D  ; *               ERROR MSG FORMATTED AND PRINTED.
              020D  ; *
              020D  ; *********************************************************
              020D  ;--
              020D  ;
              020D          $FAO_S  MSG3_ERR_CTL,FAO_LEN,FAO_DESC,TESTNUM,MSG_A
              0232                                          ; FORMAT GENERAL ERROR MSG
  01C8   30   0232          BSBW    OUTPUT_MSG              ; ... AND WRITE IT
         05   0235          RSB                             ; RETURN TO CALLER
```

```
                                  0236   383 WRITE_MSG4:
                                  0236   384 ;
                                  0236   385 ;++
                                  0236   386 ; *****************************************************
                                  0236   387 ; *
                                  0236   388 ; *      WRITE_MSG4 SUBROUTINE.  THIS ROUTINE FORMATS AND
                                  0236   389 ; *      PRINTS MESSAGES CONTAINING EXPECTED AND RECEIVED
                                  0236   390 ; *      VALUES FOR A DATA ITEM DISCOVERED TO BE IN ERROR
                                  0236   391 ; *      BY THE VERIFY SUBROUTINE.  A MESSAGE CONTAINING
                                  0236   392 ; *      THE APPROXIMATE VALUE OF PC AT THE TIME OF ERROR
                                  0236   393 ; *      IS ALSO FORMATTED AND PRINTED.
                                  0236   394 ; *
                                  0236   395 ; *      INPUTS:
                                  0236   396 ; *              MSG_CTXT:       A BYTE FIELD CONTAINING THE
                                  0236   397 ; *                              CONTEXT FOR THE DATA ITEM
                                  0236   398 ; *                              IN ERROR.
                                  0236   399 ; *
                                  0236   400 ; *              EXPV:           A QUADWORD FIELD CONTAINING
                                  0236   401 ; *                              THE EXPECTED VALUE FOR THE
                                  0236   402 ; *                              DATA ITEM IN ERROR.
                                  0236   403 ; *
                                  0236   404 ; *              RECV:           A QUADWORD FIELD CONTAINING
                                  0236   405 ; *                              THE RECEIVED VALUE FOR THE
                                  0236   406 ; *                              DATA ITEM IN ERROR.
                                  0236   407 ; *
                                  0236   408 ; *              PCV:            A LONGWORD FIELD CONTAINING
                                  0236   409 ; *                              THE VALUE OF PC AT OR SOON
                                  0236   410 ; *                              AFTER THE ERROR.
                                  0236   411 ; *
                                  0236   412 ; *      OUTPUTS:
                                  0236   413 ; *              EXPECTED/RECEIVED/PC MESSAGES FORMATTED AND PRINTED.
                                  0236   414 ; *
                                  0236   415 ; ************************************************************
                                  0236   416 ;--
                                  0236   417 ;
   000000E1'EF   000000B9'EF   DE 0236   418         MOVAL   EXP,MSG_A               ; READY "EXPECTED" TEXT FOR FAO
   000000ED'EF   000000F9'EF   DE 0241   419         MOVAL   EXPV,MSG_D              ; GET ADDR OF EXPECTED VALUE READY FOR FAO
         00'8F   000000DC'EF   91 024C   420         CMPB    MSG_CTXT,#DESC          ; IS FAILING DATA ITEM A STRING DESCRIPTOR ?
                          1E   12 0254   421         BNEQU   10$                     ; NO -- CONTINUE
                        00AE   30 0256   422         BSBW    WRITE_MSG4_1            ; YES -- FORMAT & WRITE EXPECTED STRING
   000000E1'EF   000000C2'EF   DE 0259   423         MOVAL   REC,MSG_A               ; READY "RECEIVED" TEXT FOR FAO
   000000ED'EF   00000101'EF   DE 0264   424         MOVAL   RECV,MSG_D              ; GET ADDR OF RECEIVED VALUE READY FOR FAO
                        0095   30 026F   425         BSBW    WRITE_MSG4_1            ; FORMAT & WRITE RECEIVED STRING
                          6B   11 0272   426         BRB     40$                     ; GO DO PC MESSAGE
                                  0274   427 10$:
         00'8F   000000DC'EF   91 0274   428         CMPB    MSG_CTXT,#QUAD          ; DOES FAILING DATA ITEM HAVE QUAD CONTEXT ?
                          1C   12 027C   429         BNEQU   20$                     ; NO -- HANDLE ALL OTHER CONTEXTS
           000000E5'EF    02   9A 027E   430         MOVZBL  #2,MSG_B                ; YES -- INDICATE 2 DATA ITEMS TO FAO
           000000E9'EF    09   9A 0285   431         MOVZBL  #9,MSG_C                ; ... EACH ONE 8 BYTES LONG + A BLANK
000000F1'EF   000000ED'EF 04   C1 028C   432         ADDL3   #4,MSG_D,MSG_E          ; GIVE ADDRESS OF 2ND ITEM TO FAO
                          1D   11 0298   433         BRB     30$                     ; ... AND GO FORMAT IT
                                  029A   434 20$:
           000000E5'EF    01   9A 029A   435         MOVZBL  #1,MSG_B                ; INDICATE JUST 1 DATA ITEM TO FAO
   000000E9'EF   000000DC'EF   9A 02A1   436         MOVZBL  MSG_CTXT,MSG_C          ; GET CONTEXT READY FOR FAO
   000000E9'EF   000000E9'EF   C0 02AC   437         ADDL2   MSG_C,MSG_C             ; ... DOUBLE CONTEXT TO GET PRINTABLE LENGTH
                                  02B7   438 30$:
                        0076   30 02B7   439         BSBW    WRITE_MSG4_2            ; FORMAT & PRINT EXPECTED VALUE OF DATA
```

```
     000000E1'EF    000000C2'EF    DE  02BA    440           MOVAL   REC,MSG_A              ; GIVE ADDRESS OF "RECEIVED" TEXT TO FAO
     000000ED'EF    00000101'EF    DE  02C5    441           MOVAL   RECV,MSG_D             ; GIVE ADDR OF RECEIVED VALUE TO FAO
000000F1'EF    000000ED'EF    04   C1  02D0    442           ADDL3   #4,MSG_D,MSG_E         ; GIVE ADDR OF 2ND HALF OF QUAD ITEM TO FAO
                             0051  30  02DC    443           BSBW    WRITE_MSG4_2           ; FORMAT & PRINT RECEIVED VALUE OF DATA
                                       02DF    444 40$:
     000000E1'EF    000000CB'EF    DE  02DF    445           MOVAL   APC,MSG_A              ; READY "APPROXIMATE PC" TEXT FOR FAO
              000000E5'EF     01   9A  02EA    446           MOVZBL  #1,MSG_B               ; INDICATE JUST 1 DATA ITEM TO FAO
              000000E9'EF     08   9A  02F1    447           MOVZBL  #8,MSG_C               ; ... USING 8 PRINTABLE BYTES
     000000ED'EF    00000109'EF    DE  02F8    448           MOVAL   PCV,MSG_D              ; GIVE ADDRESS OF PC VALUE TO FAO
                             002A  30  0303    449           BSBW    WRITE_MSG4_2           ; FORMAT & PRINT PC VALUE
                                       0306    450 WRITE_MSG4X:
                                   05  0306    451           RSB                            ; RETURN TO CALLER
                                       0307    452 WRITE_MSG4_1:
                                       0307    453           $FAO_S  MSG4_ERR_CTL1,FAO_LEN,FAO_DESC,MSG_A,MSG_D
                                       032C    454                                          ; FORMAT A STRING
                             00CE  30  032C    455           BSBW    OUTPUT_MSG             ; ... AND PRINT IT
                                   05  032F    456           RSB                            ; RETURN TO CALLER
                                       0330    457 WRITE_MSG4_2:
                                       0330    458           $FAO_S  MSG4_ERR_C`L2,FAO_LEN,FAO_DESC,MSG_A,   -
                                       0330    459                   MSG_B,MSG_C,@MSG_D,@MSG_E
                                       0367    460                                          ; FORMAT A NON-STRING DATA ITEM
                             0093  30  0367    461           BSBW    OUTPUT_MSG             ; ... AND PRINT IT
                                   05  036A    462           RSB                            ; RETURN TO CALLER
```

```
                                  036B  COMP_SC::
 00000101'EF    000000F9'EF   D1  036B          CMPL    EXPV,RECV                 ; IS STATUS CODE THE ONE WE EXPECTED ?
                         54   13  0376          BEQLU   COMP_SCX                  ; YES -- GO RETURN
                                  0378          ERR_EXIT   LONG,   -
                                  0378            <UNEXPECTED STATUS CODE FROM NON-SUBJECT SYSTEM SERVICE>,,,PCV
                                  03CC                                            ; NO -- ISSUE ERR_EXIT WITH PCV PARAMETER
                                  03CC  COMP_SCX:
                              05  03CC          RSB                               ; RETURN TO CALLER (SS_CHECK MACRO)
                                  03CD  ;
                                  03CD  ;
                                  03CD  PROCESS_ERR::
       0000010E'EF   01       90  03CD          MOVB    #1,EFLAG                  ; INDICATE AN ERROR HAS OCCURRED
                   007E       30  03D4          BSBW    HALT_TC                   ; HALT FURTHER EXECUTION OF TEST CASES
 0000004A'EF    00000019'EF   DE  03D7          MOVAL   TEST_MOD_FAIL,TMD_ADDR    ; INDICATE FAILED IN END MSG
00000042'EF   03   00    02   F0  03E2          INSV    #ERROR,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR ERROR
                   FE1F       30  03EB          BSBW    WRITE_MSG3                ; FORMAT & WRITE ERROR MSG
                   FE45       30  03EE          BSBW    WRITE_MSG4                ; WRITE EXP/REC/PC MSGS
            0000010D'EF       95  03F1          TSTB    CFLAG                     ; HAVE CONDITIONS ALREADY BEEN PRINTED ?
                     03       12  03F7          BNEQ    PROCESS_ERRX              ; YES -- JUST GO RETURN
                  FC04'       30  03F9          BSBW    FORM_CONDS                ; NO -- PRINT THEM BEFORE RETURNING
                                  03FC  PROCESS_ERRX:
                              05  03FC          RSB                               ; ... AND RETURN TO CALLER (ERR_EXIT MACRO)
                                  03FD  OUTPUT_MSG::
                                  03FD  ;
                                  03FD  ;   PRINT A MESSAGE FORMATTED BY FAO AND RETURN TO CALLER
                                  03FD  ;
 00000052'EF    0000004E'EF   B0  03FD          MOVW    FAO_LEN,FAO_DESC          ; SET UP ACTUAL LENGTH IN MSG BUFFER
                                  0408          PUTMSG  <#UETP$_TEXT,#1,#FAO_DESC> ; PRINT THE MSG
     00000052'EF    0082 8F   B0  041D          MOVW    #BUFFL,FAO_DESC           ; ... AND RESTORE BUFFER TO MAX LENGTH
                              05  0426          RSB
                                  0427  SAVE_REGS::
                                  0427  ;
                                  0427  ;   SAVE CONTENTS OF REGS R2 THROUGH R6 FOR USE BY CHMRTN
                                  0427  ;
         51     00000000'EF   DE  0427          MOVAL   REG_SAVE_AREA,R1          ; INITIALIZE R1 TO BEG OF AREA
                     81   52  D0  042E          MOVL    R2,(R1)+                  ; SAVE R2
                     81   53  D0  0431          MOVL    R3,(R1)+                  ; SAVE R3
                     81   54  D0  0434          MOVL    R4,(R1)+                  ; SAVE R4
                     81   55  D0  0437          MOVL    R5,(R1)+                  ; SAVE R5
                     81   56  D0  043A          MOVL    R6,(R1)+                  ; SAVE R6
                              05  043D          RSB                               ; RETURN TO CALLER
                                  043E  REST_REGS::
                                  043E  ;
                                  043E  ;   RESTORE REGS R2 THROUGH R6 ... USED IN CHMRTN AND IN MODE MACRO
                                  043E  ;
         51     00000000'EF   DE  043E          MOVAL   REG_SAVE_AREA,R1          ; INITIALIZE R1 TO BEG OF AREA
                     52   81  D0  0445          MOVL    (R1)+,R2                  ; RESTORE R2
                     53   81  D0  0448          MOVL    (R1)+,R3                  ; RESTORE R3
                     54   81  D0  044B          MOVL    (R1)+,R4                  ; RESTORE R4
                     55   81  D0  044E          MOVL    (R1)+,R5                  ; RESTORE R5
                     56   81  D0  0451          MOVL    (R1)+,R6                  ; RESTORE R6
                              05  0454          RSB                               ; RETURN TO CALLER
```

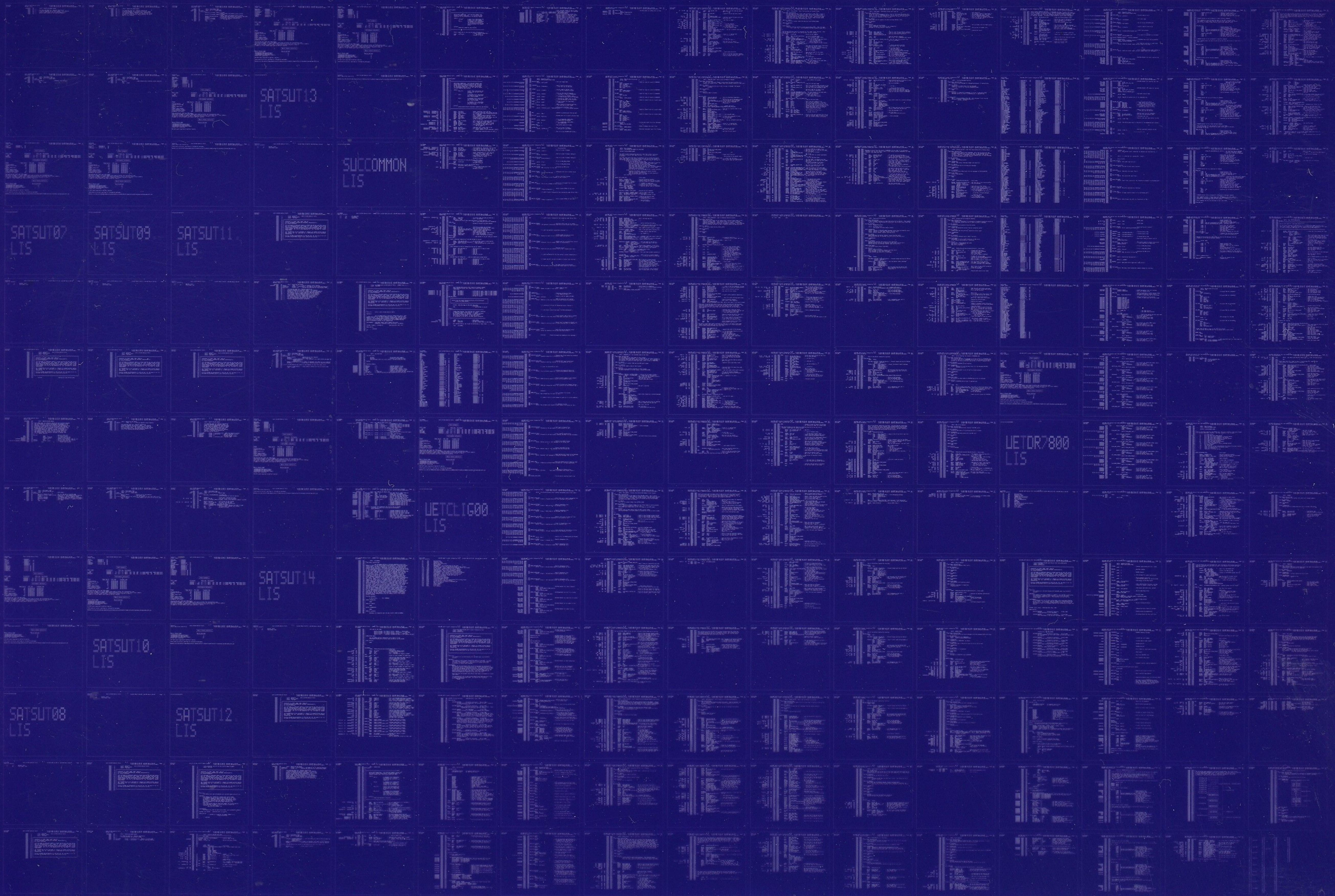
```
                          0455  HALT_TC:
                          0455  ;
                          0455  ;       HALT FURTHER EXECUTION OF TEST CASES BY STORING CURRENT
                          0455  ;       INDEX REGISTER VALUES INTO TABLE SIZE FIELDS FOR ALL
                          0455  ;       CONDITIONS.
                          0455  ;
00000000'EF   52    90    0455          MOVB    R2,COND1_H              ; ALTER END OF COND 1 TABLE TO CURR ELEMENT
00000000'EF   53    90    045C          MOVB    R3,COND2_H              ; ALTER END OF COND 2 TABLE TO CURR ELEMENT
00000000'EF   54    90    0463          MOVB    R4,COND3_H              ; ALTER END OF COND 3 TABLE TO CURR ELEMENT
00000000'EF   55    90    046A          MOVB    R5,COND4_H              ; ALTER END OF COND 4 TABLE TO CURR ELEMENT
00000000'EF   56    90    0471          MOVB    R6,COND5_H              ; ALTER END OF COND 5 TABLE TO CURR ELEMENT
                    05    0478          RSB                             ; RETURN TO CALLER
                          0479  MOD_MSG_PRINT::
                          0479  ;
                          0479  ;   ***************************************************************
                          0479  ;   *                                                             *
                          0479  ;   *  PRINTS THE TEST MODULE BEGUN/SUCCESSFUL/FAILED MESSAGES    *
                          0479  ;   *       (USING THE PUTMSG MACRO).                             *
                          0479  ;   *                                                             *
                          0479  ;   ***************************************************************
                          0479  ;
                          0479          PUTMSG  <MOD_MSG_CODE,#2,TMN_ADDR,TMD_ADDR>
                          0494                                          ; PRINT MSG ...
                    05    0494          RSB                             ; ... AND RETURN TO CALLER
                          0495  ;
                          0495  CHMRTN::
                          0495  ; ************************************************************
                          0495  ; *
                          0495  ; *     CHANGE MODE ROUTINE. THIS ROUTINE GETS CONTROL WHENEVER
                          0495  ; *     A CMKRNL OR CMEXEC SYSTEM SERVICE IS ISSUED
                          0495  ; *     BY THE MODE MACRO ('TO' OPTION).  IT MERELY RESTORES
                          0495  ; *     REGISTERS R2 THROUGH R6 AND DOES A JUMP INDIRECT
                          0495  ; *     ON A FIELD SET UP BY MODE.  IT HAS THE EFFECT
                          0495  ; *     OF RETURNING TO THE END OF THE MODE MACRO EXPANSION.
                          0495  ; *
                          0495  ; ************************************************************
                          0495  ;
                  0000    0495          .WORD   0                       ; ENTRY MASK
          FFA4      30    0497          BSBW    REST_REGS               ; RESTORE REGS R2-R6
     0000003E'FF    17    049A          JMP     @CHM_CONT               ; RETURN TO MODE MACRO IN NEW MODE
                          04A0  ;
                          04A0  ; *     RET INSTRUCTION WILL BE ISSUED IN EXPANSION OF 'MODE FROM, ....' MACRO
                          04A0  ;
                          04A0          .END    SUCCOMMON
```

```
$$$$                = 0000037A R     04      MSG4_ERR_CTL1         0000007F R     02
$$$CHARS            = 00000036               MSG4_ERR_CTL2         0000009A R     02
$$$STRINGS          = 00000001               MSG_A                 000000E1 RG    03
$$T1                = 00000004               MSG_B                 000000E5 RG    03
$$T2                = 00000008               MSG_C                 000000E9 R     03
APC                   000000CB R     02      MSG_CTXT              000000DC RG    03
BUFFL               = 00000082               MSG_D                 000000ED R     03
CFLAG                 0000010D RG    03      MSG_DATA1             000000DD RG    03
CHMRTN                00000495 RG    04      MSG_E                 000000F1 R     03
CHM_CONT              0000003E RG    03      MSG_F                 000000F5 R     03
CMPC_SAV            = 0000000C  G            NOTARG                ********   X   04
COMP_SC               0000036B RG    04      ONES                  00000000 RG    02
COMP_SCX              000003CC R     04      OUTPUT_MSG            000003FD RG    04
COND1                 ********   X   04      PCV                   00000109 RG    03
COND1_CLEANUP         ********   X   04      PROCESS_ERR           000003CD RG    04
COND1_CU              000000E3 R     04      PROCESS_ERRX          000003FC R     04
COND1_H               ********   X   04      QUAD                  ********   X   04
COND1_LOOP            00000012 R     04      R0_THRU_SP          = 00007FFF  G
COND2                 ********   X   04      REC                   000000C2 R     02
COND2_CLEANUP         ********   X   04      RECV                  00000101 RG    03
COND2_CU              000000D6 R     04      REG_SAVE_AREA         00000000 R     03
COND2_H               ********   X   04      REST_REGS             0000043E RG    04
COND2_LOOP            00000033 R     04      SAVE_REGS             00000427 RG    04
COND3                 ********   X   04      SEVERE              = 00000004
COND3_CLEANUP         ********   X   04      SHR$K_SHRDEF        = 00000001
COND3_CU              000000C9 R     04      SHR$_TEXT           = 00001130
COND3_H               ********   X   04      STS$V_INHIB_MSG     = 0000001C
COND3_LOOP            00000054 R     04      SUCCESS             = 00000001  G
COND4                 ********   X   04      SUCCOMMON             00000000 R     04
COND4_CLEANUP         ********   X   04      SYS$EXIT              ********  GX   04
COND4_CU              000000BC R     04      SYS$FAO               ********   X   04
COND4_H               ********   X   04      TESTNUM               0000003C RG    03
COND4_LOOP            0000006F R     04      TEST_MOD_BEG          00000008 R     02
COND5                 ********   X   04      TEST_MOD_FAIL         00000019 R     02
COND5_CLEANUP         ********   X   04      TEST_MOD_NAME         ********   X   03
COND5_CU              000000AF R     04      TEST_MOD_SUCC         0000000E RG    02
COND5_H               ********   X   04      TMD_ADDR              0000004A RG    03
COND5_LOOP            0000008A R     04      TMN_ADDR              00000046 R     03
CONFLICT              0000010F RG    03      TM_CLEANUP            ********   X   04
DESC                  ********   X   04      TM_CU                 000000F0 R     04
EFLAG                 0000010E RG    03      TM_SETUP              ********   X   04
ERROR               = 00000002               UETP$_SATSMS        = 007480D9
EXP                   000000B9 R     02      UETP$_TEXT          = 00741133
EXPV                  000000F9 RG    03      VERIFY                ********   X   04
FAO_DESC              00000052 RG    03      VFY_CLEANUP           ********   X   04
FAO_LFN               0000004E RG    03      WARNING             = 00000000
FORM_CONDS            ********   X   04      WRITE_MSG2            00000109 RG    04
HALT_TC               00000455 R     04      WRITE_MSG2X           00000209 R     04
INFO                = 00000003               WRITE_MSG3            0000020D R     04
LIB$SIGNAL            ********   X   04      WRITE_MSG4            00000236 R     04
LONG                  ********   X   04      WRITE_MSG4X           00000306 R     04
MOD_MSG_CODE          00000042 RG    03      WRITE_MSG4_1          00000307 R     04
MOD_MSG_PRINT         00000479 RG    04      WRITE_MSG4_2          00000330 R     04
MSG2_CND_CTL1         00000020 R     02
MSG2_CND_CTL2         00000035 R     02
MSG2_CND_CTL3         00000058 R     02
MSG3_ERR_CTL          ********   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
ROD_COMM                           000000DA  (   218.)  02 (   2.)  NOPIC    USR    CON    REL    LCL NOSHR NOEXE   RD   NOWRT NOVEC LONG
RWD_COMM                           00000110  (   272.)  03 (   3.)  NOPIC    USR    CON    REL    LCL NOSHR NOEXE   RD     WRT NOVEC LONG
SUCCOMMON                          000004A0  (  1184.)  04 (   4.)  NOPIC    USR    CON    REL    LCL NOSHR   EXE   RD     WRT NOVEC BYTE

                                                +------------------------+
                                                ! Performance indicators !
                                                +------------------------+

Phase                        Page faults    CPU Time       Elapsed Time
-----                        -----------    --------       ------------
Initialization                       37    00:00:00.09    00:00:00.30
Command processing                  136    00:00:00.68    00:00:03.17
Pass 1                              217    00:00:05.77    00:00:09.66
Symbol table sort                     0    00:00:00.35    00:00:00.39
Pass 2                              126    00:00:01.73    00:00:02.45
Symbol table output                  14    00:00:00.09    00:00:00.09
Psect synopsis output                 2    00:00:00.03    00:00:00.03
Cross-reference output                0    00:00:00.00    00:00:00.00
Assembler run totals                534    00:00:08.76    00:00:16.11

The working set limit was 1350 pages.
29666 bytes (58 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 265 non-local and 14 local symbols.
560 source lines were read in Pass 1, producing 22 object records in Pass 2.
23 pages of virtual memory were used to define 18 macros.

                                               +--------------------------+
                                               ! Macro library statistics !
                                               +--------------------------+

Macro library name                             Macros defined
------------------                             --------------
_$255$DUA28:[SHRLIB]UETP.MLB;1                            7
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                            0
_$255$DUA28:[SYSLIB]STARLET.MLB;2                         8
TOTALS (all libraries)                                   15

363 GETS were required to define 15 macros.

There were no errors, warnings or information messages.

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

0426 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



SATSUT13
LIS

SUCCOMMON
LIS

SATSUT07
LIS

SATSUT09
LIS

SATSUT11
LIS

UETDR7800
LIS

UETCLIG00
LIS

SATSUT14
LIS

SATSUT10
LIS

SATSUT08
LIS

SATSUT12
LIS