```
UUU           UUU  EEEEEEEEEEEEEEEE  TTTTTTTTTTTTTTTT  PPPPPPPPPPPP      SSSSSSSSSSSS  YYY           YYY
UUU           UUU  EEEEEEEEEEEEEEEE  TTTTTTTTTTTTTTTT  PPPPPPPPPPPP      SSSSSSSSSSSS  YYY           YYY
UUU           UUU  EEEEEEEEEEEEEEEE  TTTTTTTTTTTTTTTT  PPPPPPPPPPP       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
UUUUUUUUUUUUUUUUU  EEEEEEEEEEEEEEEE       TTT          PPP          SSSSSSSSSSSS              YYY
UUUUUUUUUUUUUUUUU  EEEEEEEEEEEEEEEE       TTT          PPP          SSSSSSSSSSSS              YYY
UUUUUUUUUUUUUUUUU  EEEEEEEEEEEEEEEE       TTT          PPP          SSSSSSSSSSSS              YYY
```

```
SSSSSSSS      AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS    333333     000000
SSSSSSSS      AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS    333333     000000
SS           AA    AA       TT       SS         SS         SS            33   33  00    00
SS           AA    AA       TT       SS         SS         SS            33   33  00    00
SS           AA    AA       TT       SS         SS         SS            33     33 00    0000
SS           AA    AA       TT       SS         SS         SS            33        00    0000
   SSSSSS    AA    AA       TT          SSSSSS     SSSSSS     SSSSSS          33  00  00  00
   SSSSSS    AA    AA       TT          SSSSSS     SSSSSS     SSSSSS          33  00  00  00
        SS   AAAAAAAAAA     TT               SS         SS         SS       33    0000    00
        SS   AAAAAAAAAA     TT               SS         SS         SS       33    0000    00
        SS   AA    AA       TT               SS         SS         SS  33   33  00      00
        SS   AA    AA       TT               SS         SS         SS  33   33  00      00
SSSSSSSS     AA    AA       TT       SSSSSSSS   SSSSSSSS   SSSSSSSS    333333     000000
SSSSSSSS     AA    AA       TT       SSSSSSSS   SSSSSSSS   SSSSSSSS    333333     000000
```

```
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
```

SATSSS30   SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54   VAX/VMS Macro V04-00   Page   0
Table of contents

```
0000      1             .TITLE   SATSSS30 SATS SYSTEM SERVICE TESTS  $CRELOG,$DELLOG (SUCC S.C.)
0000      2             .IDENT   'V04-000'
0000      3
0000      4  ;
0000      5  ;*****************************************************************************
0000      6  ;*                                                                           *
0000      7  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000      8  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000      9  ;*  ALL RIGHTS RESERVED.                                                     *
0000     10  ;*                                                                           *
0000     11  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000     12  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
0000     13  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000     14  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000     15  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000     16  ;*  TRANSFERRED.                                                             *
0000     17  ;*                                                                           *
0000     18  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000     19  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0000     20  ;*  CORPORATION.                                                             *
0000     21  ;*                                                                           *
0000     22  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000     23  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000     24  ;*                                                                           *
0000     25  ;*                                                                           *
0000     26  ;*****************************************************************************
0000     27  ;
0000     28
0000     29  ;++
0000     30  ; FACILITY:       SYSTST (SATS SYSTEM SERVICE TESTS)
0000     31  ;
0000     32  ; ABSTRACT:
0000     33  ;
0000     34  ;              THIS MODULE CONTAINS SUBROUTINES WHICH, WHEN LINKED
0000     35  ; WITH SUCCOMMON.OBJ, FORM TEST MODULE SATSSS30 TO TEST SUCCESSFUL
0000     36  ; OPERATION OF THE $CRELOG AND $DELLOG SYSTEM SERVICES. THE SERVICES ARE INVOKED
0000     37  ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000     38  ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000     39  ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000     40  ; CHECKING FOR AN SS$ NORMAL STATUS CODE, EXPECTED RETURN ARGUMENTS
0000     41  ; AND EXPECTED FUNCTIONALITY PERFORMED.
0000     42  ;
0000     43  ; ENVIRONMENT: USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000     44  ;              DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000     45  ;
0000     46  ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: MAR, 1977
0000     47  ;
0000     48  ; MODIFIED BY:
0000     49  ;
0000     50  ;              VERSION 1.50 : 25-MAY-79
0000     51  ;
0000     52  ; 01  LDJ  08/17/79     Added code to test for $DELLOG system service.
0000     53  ;--
```

K 6

SATSSS30   SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54   VAX/VMS Macro V04-00   Page   2
V04-000   DECLARATIONS   5-SEP-1984 04:30:27   [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
         0000     55            .SBTTL  DECLARATIONS
         0000     56  ;
         0000     57  ; INCLUDE FILES:
         0000     58  ;
         0000     59            $PRVDEF                         ; PRIVILEGE BIT DEFINITIONS
         0000     60            $PHDDEF                         ; PROCESS HEADER OFFSETS
         0000     61            $LOGDEF                         ; LOGICAL NAME TABLE DEFINITIONS
         0000     62            $PSLDEF                         ; PROCESSOR STATUS LONGWORD DEFINITIONS
         0000     63  ;
         0000     64  ; MACROS:
         0000     65  ;
         0000     66  ;
         0000     67  ; EQUATED SYMBOLS:
         0000     68  ;
         0000     69  ;
         0000     70  ; OWN STORAGE:
         0000     71  ;
```

L 6

SATSSS30
V04-000

SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00   Page   3
DECLARATIONS   5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
        00000000     73          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
            0000     74 TEST_MOD_NAME:: STRING  C,<SATSSS30>        ; TEST MODULE NAME
            0009     75 TEST_MOD_NAME_D: STRING I,<SATSSS30>        ; TEST MODULE NAME DESCRIPTOR
            0019     76 MSG1_INP_CTL:   STRING  I,<  SSCLN!4ZW: CONDITIONS:>
            0039     77                                            ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
            0039     78 MSG3_ERR_CTL:: STRING  I,< *SSCLN!4ZW: !AS>
            0051     79                                            ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
00000002    0051     80 EQLNAM:             .LONG  2               ; EQLNAM ARGUMENT FOR
00000000'   0055     81                     .ADDRESS TESTNUM       ; ... SUBJECT SYSTEM SERVICE
00000004'   0059     82 LOGNAM:             .LONG  4               ; LOGNAM ARGUMENT FOR
00000096'   005D     83                     .ADDRESS COMTN         ; ... SUBJECT SYSTEM SERVICE
```

```
        00000000        85           .PSECT   RWDATA,RD,WRT,NOEXE,LONG
00000008  0000          86 PRIVMASK:          .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000A  0008          87 RSLLEN_TLN:         .BLKW   1               ; RSLLEN ARGUMENT FOR $TRNLOG
          000A          88 RSLBUF_TLN:         STRING  0,130           ; RSLBUF ARGUMENT FOR $TRNLOG
00000095  0094          89 TABLE_TLN:          .BLKB   1               ; TABLE ARGUMENT FOR $TRNLOG
00000096  0095          90 ACMODE_TLN:         .BLKB   1               ; ACMODE ARGUMENT FOR $TRNLOG
00000000  0096          91 COMTN:              .LONG   0               ; COMPLEMENTED TESTNUM (USED AS LOGNAM)
```

```
              009A        93              .SBTTL  CONDITION TABLES
              009A        94 ;
              009A        95 ;;   ***** CONDITION TABLES FOR CRELOG SYSTEM SERVICE *****
              009A        96 ;
              009A        97              COND    1,LONG,<TBLFLG>,-
              009A        98                      <SYSTEM TABLE>,-
              009A        99                      <GROUP TABLE>,-
              009A       100                      <PROCESS TABLE>,-
              009A       101
    00000000  00D5       102              .LONG           LOG$C_SYSTEM
    00000001  00D9       103              .LONG           LOG$C_GROUP
    00000002  00DD       104              .LONG           LOG$C_PROCESS
              00E1       105 ;
              00E1       106              COND    2,NOTARG,<PREVIOUS STATUS OF LOGICAL NAME>,-
              00E1       107                      <ALREADY EXISTS>,-
              00E1       108                      <NON-EXISTENT>,-
              00E1       109
   00000000'  0126       110              .LONG           SS$_SUPERSEDE
   00000000'  012A       111              .LONG           SS$_NORMAL
              012E       112 ;
              012E       113              COND    3,LONG,<ACMODE>,-
              012E       114                      <KERNEL>,-
              012E       115                      <EXEC>,-
              012E       116                      <SUPER>,-
              012E       117                      <USER>,-
              012E       118
    00000000  015D       119              .LONG           PSL$C_KERNEL
    00000001  0161       120              .LONG           PSL$C_EXEC
    00000002  0165       121              .LONG           PSL$C_SUPER
    00000003  0169       122              .LONG           PSL$C_USER
              016D       123 ;
              016D       124              COND    4,NULL
              016E       125
              016E       126              COND    5,NULL
              016F       127
    00000000             128              .PSECT  SATSSS30,RD,WRT,EXE
```

SATSSS30
V04-000

B 7
SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00   Page   6
TM_SETUP, TM_CLEANUP   5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

**F

```
                        0000   130            .SBTTL  TM_SETUP, TM_CLEANUP
                        0000   131    ;++
                        0000   132    ; FUNCTIONAL DESCRIPTION:
                        0000   133    ;
                        0000   134    ;           TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                        0000   135    ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                        0000   136    ; TEST MODULE EXECUTION.
                        0000   137    ;
                        0000   138    ; CALLING SEQUENCE:
                        0000   139    ;
                        0000   140    ;     BSBW TM_SETUP   BSBW TM_CLEANUP
                        0000   141    ;
                        0000   142    ; INPUT PARAMETERS:
                        0000   143    ;
                        0000   144    ;     NONE
                        0000   145    ;
                        0000   146    ; IMPLICIT INPUTS:
                        0000   147    ;
                        0000   148    ;     NONE
                        0000   149    ;
                        0000   150    ; OUTPUT PARAMETERS:
                        0000   151    ;
                        0000   152    ;     NONE
                        0000   153    ;
                        0000   154    ; IMPLICIT OUTPUTS:
                        0000   155    ;
                        0000   156    ;     TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                        0000   157    ;               ALL PRIVILEGES ACQUIRED.
                        0000   158    ;
                        0000   159    ; COMPLETION CODES:
                        0000   160    ;
                        0000   161    ;     EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                        0000   162    ;
                        0000   163    ; SIDE EFFECTS:
                        0000   164    ;
                        0000   165    ;     SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                        0000   166    ;     (VIA RSB) IF ERROR ENCOUNTERED.
                        0000   167    ;
                        0000   168    ;--
                        0000   169
                        0000   170
                        0000   171
                        0000   172   TM_SETUP::
               52  D4   0000   173            CLRL    R2                        ; INITIALIZE
               53  D4   0002   174            CLRL    R3                        ; .. CONDITION
               54  D4   0004   175            CLRL    R4                        ; .... TABLE
               55  D4   0006   176            CLRL    R5                        ; ...... INDEX
               56  D4   0008   177            CLRL    R6                        ; ........ REGISTERS
             FFF3'  30   000A  178            BSBW    MOD_MSG_PRINT             ; PRINT TEST MODULE BEGIN MSG
   00000000'EF  00000000'EF  DE  000D  179    MOVAL   TEST_MOD_SUCC,TMD_ADDR   ; ASSUME END MSG WILL SHOW SUCCESS
      03  00  00000000'8F  F0  0018  180    INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
             00000000'EF      0020
                             0025   181            MODE    TO,5$,KRNL               ; KERNEL MODE TO ACCESS PHD
       59  00000000'9F  D0  0048  182    MOVL    @#CTL$GL_PHD,R9          ; GET PROCESS HEADER ADDRESS
      00000000'EF  69  DE  004F  183    MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                             0056   184            MODE    FROM,5$ ; BACK TO USER MODE
                             0057   185            PRIV    ADD,ALL                  ; GET ALL PRIVILEGES
```

C 7

SATSSS30   SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00   Page 7
V04-000   TM_SETUP, TM_CLEANUP   5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
                   0077   186              $SETPRN_S TEST_MOD_NAME_D        ; SET PROCESS NAME
                   0084   187              SS_CHECK  NORMAL                 ; CHECK STATUS CODE RETURNED FROM SETPRN
            05     00B2   188              RSB                              ; RETURN TO MAIN ROUTINE
                   00B3   189 TM_CLEANUP::
    FF4A'   30     00B3   190              BSBW      MOD_MSG_PRINT          ; PRINT TEST MODULE END MSG
            05     00B6   191              RSB                              ; RETURN TO MAIN ROUTINE
```

SATSSS30
V04-000

D 7
SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00 Page 8
CONDITION SUBROUTINES - SETUP AND CLEANU 5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
00B7   193                .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
00B7   194  ;++
00B7   195  ; FUNCTIONAL DESCRIPTION:
00B7   196  ;
00B7   197  ;          CONDX AND CONDX_CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
00B7   198  ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
00B7   199  ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
00B7   200  ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
00B7   201  ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
00B7   202  ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
00B7   203  ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
00B7   204  ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
00B7   205  ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
00B7   206  ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
00B7   207  ;
00B7   208  ; CALLING SEQUENCE:
00B7   209  ;
00B7   210  ;      BSBW CONDX   BSBW CONDX_CLEANUP
00B7   211  ;         WHERE X = 1,2,3,4,5
00B7   212  ;
00B7   213  ; INPUT PARAMETERS:
00B7   214  ;
00B7   215  ;      CONFLICT = 0
00B7   216  ;
00B7   217  ; IMPLICIT INPUTS:
00B7   218  ;
00B7   219  ;      R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
00B7   220  ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
00B7   221  ;
00B7   222  ; OUTPUT PARAMETERS:
00B7   223  ;
00B7   224  ;      CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
00B7   225  ;
00B7   226  ; IMPLICIT OUTPUTS:
00B7   227  ;
00B7   228  ;      R2,3,4,5,6 PRESERVED
00B7   229  ;
00B7   230  ; COMPLETION CODES:
00B7   231  ;
00B7   232  ;      NONE
00B7   233  ;
00B7   234  ; SIDE EFFECTS:
00B7   235  ;
00B7   236  ;      NONE
00B7   237  ;
00B7   238  ;--
00B7   239
00B7   240
00B7   241
00B7   242  COND1::
05   00B7   243          RSB                          ; RETURN TO MAIN ROUTINE
00B8   244  COND1_CLEANUP::
05   00B8   245          RSB                          ; RETURN TO MAIN ROUTINE
00B9   246  COND2::
05   00B9   247          RSB                          ; RETURN TO MAIN ROUTINE
00BA   248  COND2_CLEANUP::
05   00BA   249          RSB                          ; RETURN TO MAIN ROUTINE
```

SATSSS30
V04-000

SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00   Page 9
CONDITION SUBROUTINES - SETUP AND CLEANU 5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
        00BB   250 COND3::
05      00BB   251        RSB                              ; RETURN TO MAIN ROUTINE
        00BC   252 COND3_CLEANUP::
05      00BC   253        RSB                              ; RETURN TO MAIN ROUTINE
        00BD   254 COND4::
05      00BD   255        RSB                              ; RETURN TO MAIN ROUTINE
        00BE   256 COND4_CLEANUP::
05      00BE   257        RSB                              ; RETURN TO MAIN ROUTINE
        00BF   258 COND5::
05      00BF   259        RSB                              ; RETURN TO MAIN ROUTINE
        00C0   260 COND5_CLEANUP::
05      00C0   261        RSB                              ; RETURN TO MAIN ROUTINE
```

```
                          00C1      263              .SBTTL  FORM_CONDS
                          00C1      264       ;++
                          00C1      265       ; FUNCTIONAL DESCRIPTION:
                          00C1      266       ;
                          00C1      267       ;               FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                          00C1      268       ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                          00C1      269       ;
                          00C1      270       ; CALLING SEQUENCE:
                          00C1      271       ;
                          00C1      272       ;       BSBW FORM_CONDS
                          00C1      273       ;
                          00C1      274       ; INPUT PARAMETERS:
                          00C1      275       ;
                          00C1      276       ;       NONE
                          00C1      277       ;
                          00C1      278       ; IMPLICIT INPUTS:
                          00C1      279       ;
                          00C1      280       ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                          00C1      281       ;          FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                          00C1      282       ;       FOR X = 1,2,3,4,5 :
                          00C1      283       ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                          00C1      284       ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                          00C1      285       ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                          00C1      286       ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                          00C1      287       ;
                          00C1      288       ; OUTPUT PARAMETERS:
                          00C1      289       ;
                          00C1      290       ;       NONE
                          00C1      291       ;
                          00C1      292       ; IMPLICIT OUTPUTS:
                          00C1      293       ;
                          00C1      294       ;       NONE
                          00C1      295       ;
                          00C1      296       ; COMPLETION CODES:
                          00C1      297       ;
                          00C1      298       ;       NONE
                          00C1      299       ;
                          00C1      300       ; SIDE EFFECTS:
                          00C1      301       ;
                          00C1      302       ;       NONE
                          00C1      303       ;
                          00C1      304       ;--
                          00C1      305
                          00C1      306
                          00C1      307
                          00C1      308       FORM_CONDS::
                          00C1      309              $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                          00E0      310                                        ; FORMAT CONDITIONS HEADER MSG
            FF1D'   30    00E0      311              BSBW    OUTPUT_MSG        ; ... AND PRINT IT
         14   04   91    00E3      312              CMPB    #COND1_C,#NULL    ; IS CONDITION 1 NULL ?
              03   12    00E6      313              BNEQU   10$               ; NO -- CONTINUE
              00D7  31    00E8      314              BRW     FORM_CONDSX       ; YES -- SUBROUTINE IS FINISHED
                          00EB      315       10$:
00000000'EF    0000009A'EF  DE  00EB  316              MOVAL   COND1_T,MSG_A     ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF    000000A2'EF42 D0  00F6  317              MOVL    COND1_TAB[R2],MSG_B ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
00000000'EF    04   90    0102      318              MOVB    #COND1_C,MSG_CTXT  ; SAVE CONDITION 1 CONTEXT FOR FAO
                          0109      319              MOV_VAL COND1_C,CONDT_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                        FEE8'  30  0115  320          BSBW    WRITE_MSG2             ; FORMAT AND WRITE CONDITION 1 MSG
                    14    00  91  0118  321          CMPB    #COND2_C,#NULL         ; IS CONDITION 2 NULL ?
                          03  12  011B  322          BNEQU   20$                    ; NO -- CONTINUE
                        00A2  31  011D  323          BRW     FORM_CONDSX            ; YES -- SUBROUTINE IS FINISHED
                              0120  324  20$:
    00000000'EF   000000E1'EF  DE  0120  325          MOVAL   COND2_T,MSG_A          ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
    00000000'EF   00000102'EF43  D0  012B  326          MOVL    COND2_TAB[R3],MSG_B    ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
    00000000'EF        00  90  0137  327          MOVB    #COND2_C,MSG_CTXT      ; SAVE CONDITION 2 CONTEXT FOR FAO
                              013E  328          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                        FEBF'  30  013E  329          BSBQ    WRITE_MSG2             ; FORMAT AND WRITE CONDITION 2 MSG
                    14    04  91  0141  330          CMPB    #COND3_C,#NULL         ; IS CONDITION 3 NULL ?
                          03  12  0144  331          BNEQU   30$                    ; NO -- CONTINUE
                        0079  31  0146  332          BRW     FORM_CONDSX            ; YES -- SUBROUTINE IS FINISHED
                              0149  333  30$:
    00000000'EF   0000012E'EF  DE  0149  334          MOVAL   COND3_T,MSG_A          ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
    00000000'EF   00000136'EF44  D0  0154  335          MOVL    COND3_TAB[R4],MSG_B    ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
    00000000'EF        04  90  0160  336          MOVB    #COND3_C,MSG_CTXT      ; SAVE CONDITION 3 CONTEXT FOR FAO
                              0167  337          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                        FE8A'  30  0173  338          BSBQ    WRITE_MSG2             ; FORMAT AND WRITE CONDITION 3 MSG
                    14    14  91  0176  339          CMPB    #COND4_C,#NULL         ; IS CONDITION 4 NULL ?
                          47  13  0179  340          BEQLU   FORM_CONDSX            ; YES -- SUBROUTINE IS FINISHED
    00000000'EF   0000016D'EF  DE  017B  341          MOVAL   COND4_T,MSG_A          ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
    00000000'EF   0000016D'EF45  D0  0186  342          MOVL    COND4_TAB[R5],MSG_B    ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
    00000000'EF        14  90  0192  343          MOVB    #COND4_C,MSG_CTXT      ; SAVE CONDITION 4 CONTEXT FOR FAO
                              0199  344          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                        FE64'  30  0199  345          BSBQ    WRITE_MSG2             ; FORMAT AND WRITE CONDITION 4 MSG
                    14    14  91  019C  346          CMPB    #COND5_C,#NULL         ; IS CONDITION 5 NULL ?
                          21  13  019F  347          BEQLU   FORM_CONDSX            ; YES -- SUBROUTINE IS FINISHED
    00000000'EF   0000016E'EF  DE  01A1  348          MOVAL   COND5_T,MSG_A          ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
    00000000'EF   0000016E'EF46  D0  01AC  349          MOVL    COND5_TAB[R6],MSG_B    ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
    00000000'EF        14  90  01B8  350          MOVB    #COND5_C,MSG_CTXT      ; SAVE CONDITION 5 CONTEXT FOR FAO
                              01BF  351          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                        FE3E'  30  01BF  352          BSBQ    WRITE_MSG2             ; FORMAT AND WRITE CONDITION 5 MSG
                              01C2  353  FORM_CONDSX:
                          05  01C2  354          RSB                           ; RETURN TO CALLER
```

H 7

SATSSS30
V04-000

SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00 Page 12
VERIFY   5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
01C3    356        .SBTTL  VERIFY
01C3    357 ;++
01C3    358 ; FUNCTIONAL DESCRIPTION:
01C3    359 ;
01C3    360 ;        VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
01C3    361 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
01C3    362 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
01C3    363 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
01C3    364 ; ($CRELOG). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
01C3    365 ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
01C3    366 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
01C3    367 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
01C3    368 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
01C3    369 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
01C3    370 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
01C3    371 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
01C3    372 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
01C3    373 ;
01C3    374 ; CALLING SEQUENCE:
01C3    375 ;
01C3    376 ;    BSBW VERIFY
01C3    377 ;
01C3    378 ; INPUT PARAMETERS:
01C3    379 ;
01C3    380 ;    NONE
01C3    381 ;
01C3    382 ; IMPLICIT INPUTS:
01C3    383 ;
01C3    384 ;    R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
01C3    385 ;        FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
01C3    386 ;    FOR X = 1,2,3,4,5 :
01C3    387 ;        CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
01C3    388 ;            TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
01C3    389 ;            ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
01C3    390 ;            FOR CONDX_E.
01C3    391 ;
01C3    392 ; OUTPUT PARAMETERS:
01C3    393 ;
01C3    394 ;    NONE
01C3    395 ;
01C3    396 ; IMPLICIT OUTPUTS:
01C3    397 ;
01C3    398 ;    VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
01C3    399 ;    IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
01C3    400 ;    ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
01C3    401 ;    AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
01C3    402 ;    ERRORS.
01C3    403 ;
01C3    404 ; COMPLETION CODES:
01C3    405 ;
01C3    406 ;    EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
01C3    407 ;
01C3    408 ; SIDE EFFECTS:
01C3    409 ;
01C3    410 ;    SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
01C3    411 ;    (VIA RSB) IF ERROR ENCOUNTERED.
01C3    412 ;
```

I 7

SATSSS30   SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54   VAX/VMS Macro V04-00   Page 13
V04-000   VERIFY   5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
                                       01C3    413  ;--
                                       01C3    414
                                       01C3    415
                                       01C3    416
                                       01C3    417  VERIFY::
            00000000'EF         95     01C3    418         TSTB     CFLAG                          ; SHOULD CONDITIONS BE PRINTED ?
                     03         13     01C9    419         BEQL     5$                             ; NO -- CONTINUE
                   FEF3         30     01CB    420         BSBW     FORM_CONDS                     ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                                       01CE    421  5$:
     00000096'EF    00000000'EF  B2    01CE    422         MCOMW    TESTNUM,COMTN                  ; GET A LOGICAL NAME UNIQUE TO THIS T.C.
     00000126'EF43  00000000'8F  D1    01D9    423         CMPL     #SS$_NORMAL,COND2_E[R3]         ; IS NORMAL EXPECTED ?
                     03         12     01E5    424         BNEQU    25$                            ; NO -- CONTINUE
                   0073         31     01E7    425         BRW      20$                            ; YES -- GO RIGHT TO SUBJECT SERVICE
                                       01EA    426  25$:
                                       01EA    427         MODE     TO,10$,KRNL                    ; TO KERNEL FOR EXTRA CRELOG
                                       020D    428         $CRELOG_S TBLFLG[R2],LOGNAM,EQLNAM,ACMODE[R4]
                                       022E    429                                                 ; CREATE ''ALREADY EXISTENT'' LOGICAL NAME
                                       022E    430         MODE     FROM,10$                       ; BACK TO USER MODE
                                       022F    431         SS_CHECK NORMAL                         ; CHECK FOR NORMAL STATUS CODE
                                       025D    432  20$:
                                       025D    433         MODE     TO,30$,KRNL                    ; GET KERNEL FOR SUBJECT SERVICE
                                       0280    434  ;
                                       0280    435  ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                                       0280    436  ;
                                       0280    437         $CRELOG_S TBLFLG[R2],LOGNAM,EQLNAM,ACMODE[R4]
                                       02A1    438         MODE     FROM,30$                       ; BACK TO USER
     00000000'EF    00000126'EF43  D0  02A2    439         MOVL     COND2_E[R3],EXPV               ; LOAD UP EXPECTED STATUS CODE
            00000000'EF         50 D1  02AE    440         CMPL     R0,EXPV                        ; CODE RECEIVED = CODE EXPECTED ?
                     56         13     02B5    441         BEQLU    40$                            ; YES -- DO SOME MORE VERIFYING
            00000000'EF         50 D0  02B7    442         MOVL     R0,RECV                        ; GET REC'D STAT CODE INTO STORAGE
                                       02BE    443         ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM CRELOG>
                                       030D    444                                                 ; PRINT ERROR MSG & EXIT SUBROUTINE
                                       030D    445  40$:
                                       030D    446         $TRNLOG_S  LOGNAM,RSLLEN_TLN,RSLBUF_TLN,TABLE_TLN,ACMODE_TLN
                                       0334    447         SS_CHECK NORMAL                         ; CHECK FOR NORMAL STATUS CODE
     00000000'EF    000000D5'EF42  D0  0362    448         MOVL     TBLFLG[R2],EXPV                ; GET EXPECTED VALUE OUT OF COND TABLE
     00000000'EF    00000094'EF   91  036E    449         CMPB     TABLE_TLN,EXPV                 ; DID TRNLOG RETURN CORRECT TABLE FLAG VAL ?
                     54         13     0379    450         BEQLU    60$                            ; YES -- MORE VERIFYING
     00000000'EF    00000094'EF   90  037B    451         MOVB     TABLE_TLN,RECV                 ; PROCESS ERROR & EXIT ...
                                       0386    452         ERR_EXIT BYTE,<LOGICAL NAME CREATED FOR WRONG TABLE>
                                       03CF    453  60$:
            02       000000D5'EF42  D1  03CF    454         CMPL     TBLFLG[R2],#LOG$C_PROCESS  ; IS LOG NAME IN PROCESS TABLE ?
                     03         13     03D7    455         BEQLU    65$                            ; YES -- CONTINUE
                   0073         31     03D9    456         BRW      70$                            ; NO -- BYPASS ACMODE TEST
                                       03DC    457  65$:
     00000000'EF    0000015D'EF44  D0  03DC    458         MOVL     ACMODE[R4],EXPV                ; YES -- GET EXP ACMODE OUT OF COND TABLE
     00000000'EF    00000095'EF   91  03E8    459         CMPB     ACMODE_TLN,EXPV                ; DID TRNLOG RETURN CORRECT ACCESS MODE ?
                     5A         13     03F3    460         BEQLU    70$                            ; YES -- KEEP GOING
     00000000'EF    00000095'EF   90  03F5    461         MOVB     ACMODE_TLN,RECV                ; NO -- ESTAB RECV & TAKE ERROR EXIT ...
                                       0400    462         ERR_EXIT BYTE,<LOGICAL NAME CREATED FOR WRONG >, -
                                       0400    463                  <ACCESS MODE>
                                       044F    464  70$:
     00000051'EF    00000008'EF   B1  044F    465         CMPW     RSLLEN_TLN,EQLNAM              ; DID TRNLOG RETURN CORRECT STRING LENGTH ?
                     03         12     045A    466         BNEQU    75$                            ; NO -- CONTINUE
                   0068         31     045C    467         BRW      80$                            ; YES -- DO ANOTHER VERIFY
                                       045F    468  75$:
     00000000'EF    00000051'EF   B0  045F    469         MOVW     EQLNAM,EXPV                    ; LOAD UP EXPECTED AND
```

```
00000000'EF    00000008'EF   B0   046A   470            MOVW    RSLLEN_TLN,RECV          ; ... RECEIVED VALUES, THEN EXIT
                                   0475   471            ERR_EXIT WORD,<INCORRECT LENGTH CREATED FOR >, -
                                   0475   472                    <EQUIVALENCE NAME>
                                   04C7   473 80$:
                   0000'8F   BB   04C7   474            PUSHR   #CMPC_SAV                ; SAVE SOME REGS USED BY CMPC
0000000E'FF    00000008'EF   29   04CB   475            CMPC    RSLLEN_TLN,@RSLBUF_TLN+4,@EQLNAM+4
               00000055'FF        04D6
                                   04DB   476                                            ; TRANSLATED STRING MATCH THAT CREATED ?
                   0000'8F   BA   04DB   477            POPR    #CMPC_SAV                ; RESTORE SOME REGS USED BY CMPC
                         03   12   04DF   478            BNEQU   85$                      ; NO -- CONTINUE
                       0064   31   04E1   479            BRW     VERIFYX                  ; YES -- EVERYTHING VERIFIES
                                   04E4   480 85$:
00000000'EF    00000051'EF   7D   04E4   481            MOVQ    EQLNAM,EXPV              ; LOAD UP EXPECTED AND
00000000'EF    0000000A'EF   7D   04EF   482            MOVQ    RSLBUF_TLN,RECV          ; ... RECEIVED VALUES, THEN EXIT
                                   04FA   483            ERR_EXIT DESC,<INCORRECT EQUIVALENCE NAME >, -
                                   04FA   484                    <STRING CREATED>
                                   0548   485 VERIFYX:
                         05        0548   486            RSB                              ; RETURN TO CALLER
```

```
0549    488                         .SBTTL  VFY_CLEANUP
0549    489    ;++
0549    490    ; FUNCTIONAL DESCRIPTION:
0549    491    ;
0549    492    ;               VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
0549    493    ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
0549    494    ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
0549    495    ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
0549    496    ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
0549    497    ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
0549    498    ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
0549    499    ; POSSIBLY DISCOVERING A SECOND ERROR.
0549    500    ;
0549    501    ; CALLING SEQUENCE:
0549    502    ;
0549    503    ;       BSBW VFY_CLEANUP
0549    504    ;
0549    505    ; INPUT PARAMETERS:
0549    506    ;
0549    507    ;       NONE
0549    508    ;
0549    509    ; IMPLICIT INPUTS:
0549    510    ;
0549    511    ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
0549    512    ;          FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
0549    513    ;       FOR X = 1,2,3,4,5 :
0549    514    ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
0549    515    ;                  TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
0549    516    ;                  ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
0549    517    ;                  FOR CONDX_E.
0549    518    ;
0549    519    ; OUTPUT PARAMETERS:
0549    520    ;
0549    521    ;       NONE
0549    522    ;
0549    523    ; IMPLICIT OUTPUTS:
0549    524    ;
0549    525    ;       NONE
0549    526    ;
0549    527    ; COMPLETION CODES:
0549    528    ;
0549    529    ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
0549    530    ;
0549    531    ; SIDE EFFECTS:
0549    532    ;
0549    533    ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
0549    534    ;       (VIA RSB) IF ERROR ENCOUNTERED.
0549    535    ;
0549    536    ;--
0549    537
0549    538
0549    539
0549    540    VFY_CLEANUP::
0549    541            MODE    TO,10$,KRNL                  ; KERNEL MODE
056C    542            $DELLOG_S TBLFLG[R2],LOGNAM,ACMODE[R4] ; UNDO SUBJECT SERVICE
0587    543            MODE    FROM,10$                     ; BACK TO USER MODE
0588    544            SS_CHECK NORMAL                      ; CHECK NORMAL STATUS CODE
```

L 7

SATSSS30   SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00   Page 16
V04-000   VFY_CLEANUP   5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

```
        05B6   545          MODE    TO,20$,KRNL                         ; kernel mode
        05D9   546          $DELLOG_S TBLFLG[R2],LOGNAM,ACMODE[R4] ; check for actual delete
        05F4   547          MODE    FROM,20$                            ; back to user mode
        05F5   548          SS_CHECK NOLOGNAM                           ; check correct status return
     05 0623   549          RSB                                         ; RETURN TO CALLER
        0624   550          .END
```

Symbol	Value	Flags		Symbol	Value	Flags	
$$$$	= 00000504	R	04	EQLNAM	00000051	R	02
$$$CHARS	= 00000029			EXPV	********	X	04
$$$CHARS1	= 00000006			FAO_DESC	********	X	04
$$$CHARS2	= 00000004			FAO_LEN	********	X	04
$$$CHARS3	= 00000005			FORM_CONDS	000000C1	RG	04
$$$CHARS4	= 00000004			FORM_CONDSX	000001C2	R	04
$$$CHARS5	= 00000000			LOG$C_GROUP	= 00000001		
$$$COND_A	= 00000003			LOG$C_PROCESS	= 00000002		
$$$STRINGS	= 00000001			LOG$C_SYSTEM	= 00000000		
$$$STRINGS2	= 00000005			LOGNAM	00000059	R	02
$$T1	= 00000000			LONG	= 00000004	G	
$$T2	= 00000004			MOD_MSG_CODE	********	X	04
ACMODE	0000015D	R	03	MOD_MSG_PRINT	********	X	04
ACMODE_TLN	00000095	R	03	MSGT_INP_CTL	00000019	R	02
BYTE	= 00000001	G		MSG3_ERR_CTL	00000039	RG	02
CFLAG	********	X	04	MSG_A	********	X	04
CHMRTN	********	X	04	MSG_B	********	X	04
CHM_CONT	********	X	04	MSG_CTXT	********	X	04
CMPC_SAV	********	X	04	MSG_DATA1	********	X	04
COMP_SC	********	X	04	NOTARG	= 00000000	G	
COMTN	00000096	R	03	NULL	= 00000014	G	
COND1	000000B7	RG	04	OUTPUT_MSG	********	X	04
COND1_C	= 00000004			PCV	********	X	04
COND1_CLEANUP	000000B8	RG	04	PHD$Q_PRIVMSK	= 00000000		
COND1_E	000000D5	R	03	PRIVMASK	00000000	R	03
COND1_H	000000A1	RG	03	PRIV_ARGS	= 00000002		
COND1_T	0000009A	R	03	PROCESS_ERR	********	X	04
COND1_TAB	000000A2	R	03	PSL$C_EXEC	= 00000001		
COND2	000000B9	RG	04	PSL$C_KERNEL	= 00000000		
COND2_C	= 00000000			PSL$C_SUPER	= 00000002		
COND2_CLEANUP	000000BA	RG	04	PSL$C_USER	= 00000003		
COND2_E	00000126	R	03	QUAD	= 00000008	G	
COND2_H	00000101	RG	03	RECV	********	X	04
COND2_T	000000E1	R	03	REST_REGS	********	X	04
COND2_TAB	00000102	R	03	RSLBUF_TLN	0000000A	R	03
COND3	000000BB	RG	04	RSLLEN_TLN	00000008	R	03
COND3_C	= 00000004			SAVE_REGS	********	X	04
COND3_CLEANUP	000000BC	RG	04	SS$_NOLOGNAM	********	X	04
COND3_E	0000015D	R	03	SS$_NORMAL	********	X	03
COND3_H	00000135	RG	03	SS$_SUPERSEDE	********	X	03
COND3_T	0000012E	R	03	SUCCESS	********	X	04
COND3_TAB	00000136	R	03	SYS$CMKRNL	********	GX	04
COND4	000000BD	RG	04	SYS$CRELOG	********	GX	04
COND4_C	= 00000014			SYS$DELLOG	********	GX	04
COND4_CLEANUP	000000BE	RG	04	SYS$FAO	********	X	04
COND4_H	0000016D	RG	03	SYS$SETPRN	********	GX	04
COND4_T	0000016D	R	03	SYS$SETPRV	********	GX	04
COND4_TAB	0000016D	R	03	SYS$TRNLOG	********	GX	04
COND5	000000BF	RG	04	TABLE_TLN	00000094	R	03
COND5_C	= 00000014			TBLFLG	000000D5	R	03
COND5_CLEANUP	000000C0	RG	04	TESTNUM	********	X	02
COND5_H	0000016E	RG	03	TEST_MOD_NAME	00000000	RG	02
COND5_T	0000016E	R	03	TEST_MOD_NAME_D	00000009	R	02
COND5_TAB	0000016E	R	03	TEST_MOD_SUCC_	********	X	04
CTL$GL_PHD	********	X	04	TMD_ADDR	********	X	04
DESC	= 00000010	G		TM_CLEANUP	000000B3	RG	04
EFLAG	********	X	04	TM_SETUP	00000000	RG	04

SATSSS30   SATS SYSTEM SERVICE TESTS $CRELOG,$DELL 16-SEP-1984 00:49:54 VAX/VMS Macro V04-00   Page 18
Symbol table   5-SEP-1984 04:30:27 [UETPSY.SRC]SATSSS30.MAR;1   (1)

VERIFY   000001C3 RG   04
VERIFYX   00000548 R   04
VFY_CLEANUP   00000549 RG   04
WORD   = 00000002 G
WRITE_MSG2   ******** X   04

                                    +------------------+
                                    ! Psect synopsis !
                                    +------------------+

PSECT name   Allocation   PSECT No.   Attributes
----------   ----------   --------   ----------
. ABS .   00000000 (0.)   00 (0.)   NOPIC USR CON ABS LCL NOSHR NOEXE NORD   NOWRT NOVEC BYTE
ABS 00000000 (0.)   01 (1.)   NOPIC USR CON ABS LCL NOSHR EXE   RD   WRT NOVEC BYTE
RODATA   00000061 (97.)   02 (2.)   NOPIC USR CON REL LCL NOSHR NOEXE RD   NOWRT NOVEC LONG
RWDATA   0000016F (367.)   03 (3.)   NOPIC USR CON REL LCL NOSHR NOEXE RD   WRT NOVEC LONG
SATSSS30   00000624 (1572.)   04 (4.)   NOPIC USR CON REL LCL NOSHR EXE   RD   WRT NOVEC BYTE

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

Phase   Page faults   CPU Time   Elapsed Time
-----   -----------   --------   ------------
Initialization   29   00:00:00.07   00:00:00.26
Command processing   107   00:00:00.68   00:00:03.12
Pass 1   265   00:00:07.76   00:00:13.71
Symbol table sort   0   00:00:00.54   00:00:00.58
Pass 2   118   00:00:02.00   00:00:02.44
Symbol table output   15   00:00:00.09   00:00:00.13
Psect synopsis output   2   00:00:00.03   00:00:00.04
Cross-reference output   0   00:00:00.00   00:00:00.00
Assembler run totals   538   00:00:11.20   00:00:20.29

The working set limit was 1350 pages.
40009 bytes (79 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 361 non-local and 47 local symbols.
550 source lines were read in Pass 1, producing 25 object records in Pass 2.
37 pages of virtual memory were used to define 28 macros.

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

Macro library name   Macros defined
------------------   --------------
-255DUA28:[SHRLIB]UETP.MLB;1   9
-255DUA28:[SYS.OBJ]LIB.MLB;1   2
-255DUA28:[SYSLIB]STARLET.MLB;2   14
TOTALS (all libraries)   25

687 GETS were required to define 25 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:SATSSS30/OBJ=OBJ$:SATSSS30 MSRC$:SATSSS30/UPDATE=(ENH$:SATSSS30)+EXECML$/LIB+SHRLIB$:UETP/LIB