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

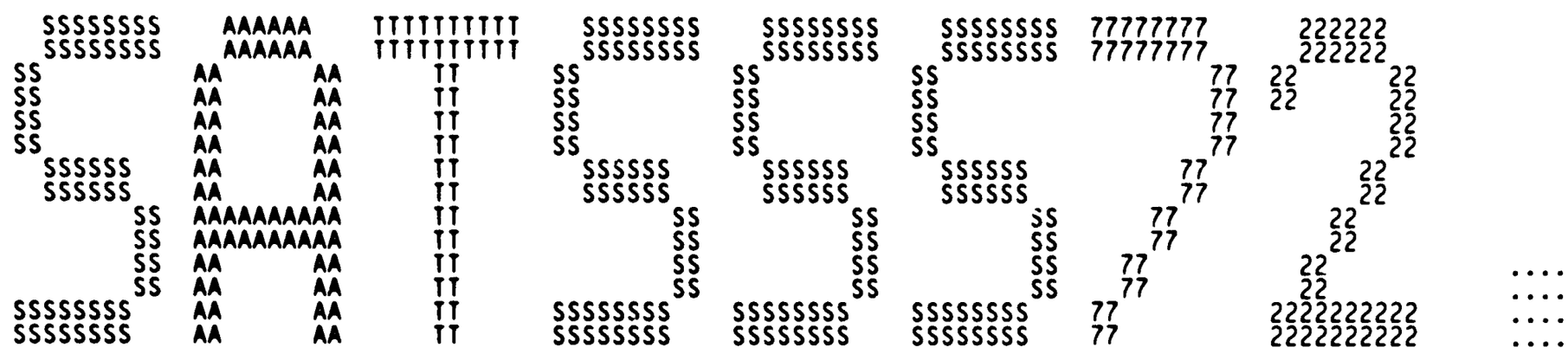
SSSSSSSS AAAAAA TTTTTTTTTT SSSSSSSS SSSSSSSS SSSSSSSS 77777777 222222
SSSSSSSS AAAAAA TTTTTTTTTT SSSSSSSS SSSSSSSS SSSSSSSS 77777777 222222
SS AA AA TT SS SS SS 77 22 22
SS AA AA TT SS SS SS 77 22 22
SS AA AA TT SS SS SS 77 22
SS AA AA TT SS SS SS 77 22
SSSSSS AA AA TT SSSSSS SSSSSS SSSSSS 77 22
SSSSSS AA AA TT SSSSSS SSSSSS SSSSSS 77 22
SS AAAAAAAAAA TT SS SS SS 77 22
SS AAAAAAAAAA TT SS SS SS 77 22
SS AA AA TT SS SS SS 77 22
SS AA AA TT SS SS SS 77 22
SSSSSSSS AA AA TT SSSSSSSS SSSSSSSS SSSSSSSS 77 2222222222
SSSSSSSS AA AA TT SSSSSSSS SSSSSSSS SSSSSSSS 77 2222222222
LL IIIIII SSSSSSSS
LL IIIIII SSSSSSSS
LL II SS
LL II SS
LL II SS
LL II SS
LL II SSSSSS
LL II SSSSSS
LL II SS
LL II SS
LL II SS
LL II SS
LLLLLLLLLL IIIIII SSSSSSSS
LLLLLLLLLL IIIIII SSSSSSSS

Table of contents

```
0000     1          .TITLE  SATSSS72 SATS SYSTEM SERVICE TESTS  $CRETVA (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 SATSSS72 TO TEST SUCCESSFUL
0000    36 ; OPERATION OF THE $CRETVA SYSTEM SERVICE. THE SERVICE IS INVOKED
0000    37 ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38 ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39 ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40 ; CHECKING FOR AN SS$_NORMAL STATUS CODE, EXPECTED RETURN ARGUMENTS
0000    41 ; AND EXPECTED FUNCTIONALITY PERFORMED.
0000    42 ;
0000    43 ; ENVIRONMENT:  USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000    44 ;               DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000    45 ;
0000    46 ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: JUL, 1977
0000    47 ;
0000    48 ; MODIFIED BY:
0000    49 ;
0000    50 ;          , : VERSION
0000    51 ; 01     -
0000    52 ;--
```

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

```
                00000000    71          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
                    0000    72 TEST_MOD_NAME:: STRING  C,<SATSSS72>     ; TEST MODULE NAME
                    0009    73 TEST_MOD_NAME_D: STRING I,<SATSSS72>     ; TEST MODULE NAME DESCRIPTOR
                    0019    74 MSG1_INP_CTL:   STRING  I,<  SSCVA!4ZW: CONDITIONS:>
                    0039    75                                          ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
                    0039    76 MSG3_ERR_CTL::  STRING  I,< *SSCVA!4ZW: !AS>
                    0051    77                                          ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
            00000200  0051    78 PAGESIZE:       .LONG   512             ; PAGE SIZE IN BYTES
```

```
                00000000     80          .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008  0000  00000008     81 PRIVMASK:        .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000C  0008  0000000C     82 PAGCNT_ERG:      .BLKL   1               ; PACCNT ARGUMENT FOR EXPREG
00000014  000C  00000014     83 RETADR_ERG:      .BLKQ   1               ; RETADR ARGUMENT FOR EXPREG
0000001C  0014  0000001C     84 INADR_CVA:       .BLKQ   1               ; INADR ARGUMENT FOR PRELIM CRETVA
00000024  001C  00000024     85 INADR:           .BLKQ   1               ; INADR ARGUMENT FOR SUBJECT CRETVA
0000002C  0024  0000002C     86 RETADR:          .BLKQ   1               ; RETADR ARGUMENT FOR SUBJECT CRETVA
00000030  002C  00000030     87 STARTADDR:       .BLKL   1               ; STARTING ADDR OF CREATED AREA
00000034  0030  00000034     88 ENDADDR:         .BLKL   1               ; ENDING ADDR OF CREATED AREA
00000038  0034  00000038     89 DISPL:           .BLKL   1               ; NO. OF PAGES (DISPLACEMENT) TO CREATE
00000039  0038  00000039     90 NZERR:           .BLKB   1               ; NON-ZERO ERROR FLAG: 0 MEANS NO N-Z ERROR
```

```
                                0039     92          .SBTTL  CONDITION TABLES
                                0039     93 ;
                                0039     94 ;       ***** CONDITION TABLES FOR CRETVA SYSTEM SERVICE *****
                                0039     95 ;
                                0039     96          COND    1,NOTARG,<PREVIOUS CONDITION OF CREATED PAGES>,-
                                0039     97                    <PREVIOUSLY CREATED BY CRETVA>,-
                                0039     98                    <PREVIOUSLY CREATED BY EXPREG>,-
                                0039     99                    <PREVIOUSLY NON-EXISTENT>,-
                                0039    100
                                00BC    101          COND    2,NOTARG,<ORDERING OF INADR PAIR>,-
                                00BC    102                    <1ST ADDR LESS THAN OR EQUAL TO 2ND ADDR>,-
                                00BC    103                    <1ST ADDR GREATER THAN OR EQUAL TO 2ND ADDR>,-
                                00BC    104
                                012F    105          COND    3,LONG,<ACMODE>,-
                                012F    106                    <KERNEL>,-
                                012F    107                    <EXEC>,-
                                012F    108                    <SUPER>,-
                                012F    109                    <USER>,-
                                012F    110
                    00000000    015E    111                      .LONG       PSL$C_KERNEL
                    00000001    0162    112                      .LONG       PSL$C_EXEC
                    00000002    0166    113                      .LONG       PSL$C_SUPER
                    00000003    016A    114                      .LONG       PSL$C_USER
                                016E    115 ;
                                016E    116          COND    4,NOTARG,<LOCATION OF CREATED PAGES>,-
                                016E    117                    <END OF PROGRAM REGION>,-
                                016E    118                    <END OF CONTROL REGION>,-
                                016E    119                    <MIDDLE OF PROGRAM REGION>,-
                                016E    120                    <MIDDLE OF CONTROL REGION>,-
                                016E    121
00000003 00000002 00000001 00000000 01F7    122                      .LONG       0,1,2,3
                                0207    123 ;
                                0207    124          COND    5,NOTARG,<PAGE COUNT>,-
                                0207    125                    <ONE PAGE>,-
                                0207    126                    <SMALL COUNT>,-
                                0207    127                    <LARGE COUNT>,-
                                0207    128
                    00000001    0240    129                      .LONG       1
                    00000005    0244    130                      .LONG       5
                    00000019    0248    131                      .LONG       25
                                024C    132 ;
                           00000000     133          .PSECT  SATSSS72,RD,WRT,EXE
```

```
                                0000   135          .SBTTL  TM_SETUP, TM_CLEANUP
                                0000   136 ;++
                                0000   137 ; FUNCTIONAL DESCRIPTION:
                                0000   138 ;
                                0000   139 ;               TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                                0000   140 ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                                0000   141 ; TEST MODULE EXECUTION.
                                0000   142 ;
                                0000   143 ; CALLING SEQUENCE:
                                0000   144 ;
                                0000   145 ;       BSBW TM_SETUP   BSBW TM_CLEANUP
                                0000   146 ;
                                0000   147 ; INPUT PARAMETERS:
                                0000   148 ;
                                0000   149 ;       NONE
                                0000   150 ;
                                0000   151 ; IMPLICIT INPUTS:
                                0000   152 ;
                                0000   153 ;       NONE
                                0000   154 ;
                                0000   155 ; OUTPUT PARAMETERS:
                                0000   156 ;
                                0000   157 ;       NONE
                                0000   158 ;
                                0000   159 ; IMPLICIT OUTPUTS:
                                0000   160 ;
                                0000   161 ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                                0000   162 ;                  ALL PRIVILEGES ACQUIRED.
                                0000   163 ;
                                0000   164 ; COMPLETION CODES:
                                0000   165 ;
                                0000   166 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                                0000   167 ;
                                0000   168 ; SIDE EFFECTS:
                                0000   169 ;
                                0000   170 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                                0000   171 ;       (VIA RSB) IF ERROR ENCOUNTERED.
                                0000   172 ;
                                0000   173 ;--
                                0000   174
                                0000   175
                                0000   176
                                0000   177 TM_SETUP::
                          52    D4    0000   178          CLRL    R2                      ; INITIALIZE
                          53    D4    0002   179          CLRL    R3                      ; .. CONDITION
                          54    D4    0004   180          CLRL    R4                      ; .... TABLE
                          55    D4    0006   181          CLRL    R5                      ; ...... INDEX
                          56    D4    0008   182          CLRL    R6                      ; ........ REGISTERS
                      FFF3'     30    000A   183          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE BEGIN MSG
00000000'EF     00000000'EF     DE    000D   184          MOVAL   TEST_MOD_SUCC,TMD_ADDR  ; ASSUME END MSG WILL SHOW SUCCESS
    03   00     0000'8F         F0    0018   185          INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
                00000000'EF           0020
                                      0025   186          MODE    TO,5$,KRNL              ; KERNEL MODE TO ACCESS PHD
          59    00000000'9F     D0    0048   187          MOVL    @#CTL$GL_PHD,R9         ; GET PROCESS HEADER ADDRESS
          00000000'EF   69      DE    004F   188          MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                                      0056   189          MODE    FROM,5$ ; BACK TO USER MODE
                                      0057   190          PRIV    ADD,ALL                 ; GET ALL PRIVILEGES
```

```
                     0077   191          $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                     0084   192          SS_CHECK  NORMAL                ; CHECK STATUS CODE RETURNED FROM SETPRN
              05     00B2   193          RSB                             ; RETURN TO MAIN ROUTINE
                     00B3   194 TM_CLEANUP::
          FF4A'      30     00B3   195          BSBW    MOD_MSG_PRINT   ; PRINT TEST MODULE END MSG
              05     00B6   196          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                        .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
                ;++
                ; FUNCTIONAL DESCRIPTION:
                ;
                ;               CONDX AND CONDX_CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
                ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
                ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
                ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
                ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
                ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
                ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
                ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
                ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
                ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
                ;
                ; CALLING SEQUENCE:
                ;
                ;       BSBW CONDX    BSBW CONDX_CLEANUP
                ;         WHERE X = 1,2,3,4,5
                ;
                ; INPUT PARAMETERS:
                ;
                ;       CONFLICT = 0
                ;
                ; IMPLICIT INPUTS:
                ;
                ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                ;
                ; OUTPUT PARAMETERS:
                ;
                ;       CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
                ;
                ; IMPLICIT OUTPUTS:
                ;
                ;       R2,3,4,5,6 PRESERVED
                ;
                ; COMPLETION CODES:
                ;
                ;       NONE
                ;
                ; SIDE EFFECTS:
                ;
                ;       NONE
                ;
                ;--


                COND1::
05              RSB                             ; RETURN TO MAIN ROUTINE
                COND1_CLEANUP::
05              RSB                             ; RETURN TO MAIN ROUTINE
                COND2::
05              RSB                             ; RETURN TO MAIN ROUTINE
                COND2_CLEANUP::
05              RSB                             ; RETURN TO MAIN ROUTINE
```

```
                00BB   255 COND3::
 05             00BB   256          RSB                         ; RETURN TO MAIN ROUTINE
                00BC   257 COND3_CLEANUP::
 05             00BC   258          RSB                         ; RETURN TO MAIN ROUTINE
                00BD   259 COND4::
 05             00BD   260          RSB                         ; RETURN TO MAIN ROUTINE
                00BE   261 COND4_CLEANUP::
 05             00BE   262          RSB                         ; RETURN TO MAIN ROUTINE
                00BF   263 COND5::
 05             00BF   264          RSB                         ; RETURN TO MAIN ROUTINE
                00C0   265 COND5_CLEANUP::
 05             00C0   266          RSB                         ; RETURN TO MAIN ROUTINE
```

```
                                00C1    268          .SBTTL  FORM_CONDS
                                00C1    269 ;++
                                00C1    270 ; FUNCTIONAL DESCRIPTION:
                                00C1    271 ;
                                00C1    272 ;                   FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                                00C1    273 ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                                00C1    274 ;
                                00C1    275 ; CALLING SEQUENCE:
                                00C1    276 ;
                                00C1    277 ;       BSBW FORM_CONDS
                                00C1    278 ;
                                00C1    279 ; INPUT PARAMETERS:
                                00C1    280 ;
                                00C1    281 ;       NONE
                                00C1    282 ;
                                00C1    283 ; IMPLICIT INPUTS:
                                00C1    284 ;
                                00C1    285 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                                00C1    286 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                                00C1    287 ;       FOR X = 1,2,3,4,5 :
                                00C1    288 ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                                00C1    289 ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                                00C1    290 ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                                00C1    291 ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                                00C1    292 ;
                                00C1    293 ; OUTPUT PARAMETERS:
                                00C1    294 ;
                                00C1    295 ;       NONE
                                00C1    296 ;
                                00C1    297 ; IMPLICIT OUTPUTS:
                                00C1    298 ;
                                00C1    299 ;       NONE
                                00C1    300 ;
                                00C1    301 ; COMPLETION CODES:
                                00C1    302 ;
                                00C1    303 ;       NONE
                                00C1    304 ;
                                00C1    305 ; SIDE EFFECTS:
                                00C1    306 ;
                                00C1    307 ;       NONE
                                00C1    308 ;
                                00C1    309 ;--
                                00C1    310
                                00C1    311
                                00C1    312
                                00C1    313 FORM_CONDS::
                                00C1    314         $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                                00E0    315                                             ; FORMAT CONDITIONS HEADER MSG
                          FF1D'  30     00E0    316         BSBW    OUTPUT_MSG              ; ... AND PRINT IT
                       14   00  91     00E3    317         CMPB    #COND1_C,#NULL          ; IS CONDITION 1 NULL ?
                            03  12     00E6    318         BNEQU   10$                     ; NO -- CONTINUE
                           00CB 31     00E8    319         BRW     FORM_CONDSX             ; YES -- SUBROUTINE IS FINISHED
                                00EB    320 10$:
  00000000'EF   00000039'EF   DE    00EB    321         MOVAL   COND1_T,MSG_A           ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF   0000005E'EF42   D0    00F6    322         MOVL    COND1_TAB[R2],MSG_B     ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
          00000000'EF   00    90    0102    323         MOVB    #COND1_C,MSG_CTXT       ; SAVE CONDITION 1 CONTEXT FOR FAO
                                0109    324         MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                     FEF4'  30  0109   325          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 1 MSG
                 14    00   91  010C   326          CMPB    #COND2_C,#NULL                  ; IS CONDITION 2 NULL ?
                       03   12  010F   327          BNEQU   20$                             ; NO -- CONTINUE
                    00A2    31  0111   328          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                            0114   329 20$:
  00000000'EF   000000BC'EF   DE  0114   330          MOVAL   COND2_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF   000000D4'EF43   D0  011F   331          MOVL    COND2_TAB[R3],MSG_B             ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
         00000000'EF    00   90  0128   332          MOVB    #COND2_C,MSG_CTXT               ; SAVE CONDITION 2 CONTEXT FOR FAO
                             0132   333          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                     FECB'  30  0132   334          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 2 MSG
                 14    04   91  0135   335          CMPB    #COND3_C,#NULL                  ; IS CONDITION 3 NULL ?
                       03   12  0138   336          BNEQU   30$                             ; NO -- CONTINUE
                    0079    31  013A   337          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                            013D   338 30$:
  00000000'EF   0000012F'EF   DE  013D   339          MOVAL   COND3_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF   00000137'EF44   D0  0148   340          MOVL    COND3_TAB[R4],MSG_B             ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
         00000000'EF    04   90  0154   341          MOVB    #COND3_C,MSG_CTXT               ; SAVE CONDITION 3 CONTEXT FOR FAO
                             015B   342          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                     FE96'  30  0167   343          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 3 MSG
                 14    00   91  016A   344          CMPB    #COND4_C,#NULL                  ; IS CONDITION 4 NULL ?
                       47   13  016D   345          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   0000016E'EF   DE  016F   346          MOVAL   COND4_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF   00000189'EF45   D0  017A   347          MOVL    COND4_TAB[R5],MSG_B             ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
         00000000'EF    00   90  0186   348          MOVB    #COND4_C,MSG_CTXT               ; SAVE CONDITION 4 CONTEXT FOR FAO
                             018D   349          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                     FE70'  30  018D   350          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 4 MSG
                 14    00   91  0190   351          CMPB    #COND5_C,#NULL                  ; IS CONDITION 5 NULL ?
                       21   13  0193   352          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   00000207'EF   DE  0195   353          MOVAL   COND5_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF   00000213'EF46   D0  01A0   354          MOVL    COND5_TAB[R6],MSG_B             ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
         00000000'EF    00   90  01AC   355          MOVB    #COND5_C,MSG_CTXT               ; SAVE CONDITION 5 CONTEXT FOR FAO
                             01B3   356          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                     FE4A'  30  01B3   357          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 5 MSG
                            01B6   358 FORM_CONDSX:
                        05  01B6   359          RSB                                     ; RETURN TO CALLER
```

```
01B7   361          .SBTTL  VERIFY
01B7   362 ;++
01B7   363 ; FUNCTIONAL DESCRIPTION:
01B7   364 ;
01B7   365 ;                    VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
01B7   366 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
01B7   367 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
01B7   368 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
01B7   369 ; ($CRETVA). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
01B7   370 ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
01B7   371 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
01B7   372 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
01B7   373 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
01B7   374 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
01B7   375 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
01B7   376 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
01B7   377 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
01B7   378 ;
01B7   379 ; CALLING SEQUENCE:
01B7   380 ;
01B7   381 ;       BSBW VERIFY
01B7   382 ;
01B7   383 ; INPUT PARAMETERS:
01B7   384 ;
01B7   385 ;       NONE
01B7   386 ;
01B7   387 ; IMPLICIT INPUTS:
01B7   388 ;
01B7   389 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
01B7   390 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
01B7   391 ;       FOR X = 1,2,3,4,5 :
01B7   392 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
01B7   393 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
01B7   394 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
01B7   395 ;                 FOR CONDX_E.
01B7   396 ;
01B7   397 ; OUTPUT PARAMETERS:
01B7   398 ;
01B7   399 ;       NONE
01B7   400 ;
01B7   401 ; IMPLICIT OUTPUTS:
01B7   402 ;
01B7   403 ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
01B7   404 ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
01B7   405 ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
01B7   406 ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
01B7   407 ;       ERRORS.
01B7   408 ;
01B7   409 ; COMPLETION CODES:
01B7   410 ;
01B7   411 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
01B7   412 ;
01B7   413 ; SIDE EFFECTS:
01B7   414 ;
01B7   415 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
01B7   416 ;       (VIA RSB) IF ERROR ENCOUNTERED.
01B7   417 ;
```

```
                                    01B7   418 ;--
                                    01B7   419
                                    01B7   420
                                    01B7   421
                                    01B7   422 VERIFY::
                00000000'EF   95    01B7   423          TSTB    CFLAG                    ; SHOULD CONDITIONS BE PRINTED ?
                        03    13    01BD   424          BEQL    5$                       ; NO -- CONTINUE
                      FEFF    30    01BF   425          BSBW    FORM_CONDS               ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                                    01C2   426 5$:
                0000002C'EF   D4    01C2   427          CLRL    STARTADDR                ; CLEAR STARTING ADDR OF AREA TO BE CREATED
  00000008'EF   00000240'EF46   D0    01C8   428          MOVL    COND5_E[R6],PAGCNT_ERG   ; GET PAGE COUNT
    00000034'EF   00000008'EF   D0    01D4   429          MOVL    PAGCNT_ERG,DISPL         ; ... AND REMEMBER IT FOR LATER
          01   000001F7'EF45   D1    01DF   430          CMPL    COND4_E[R5],#1           ; ARE PAGES IN MIDDLE OF A REGION ?
                        07    15    01E7   431          BLEQ    10$                      ; NO -- PAGCNT_ERG IS OK
             00000008'EF   05   C0    01F9   432          ADDL2   #5,PAGCNT_ERG            ; YES -- ADD SOME PAGES
                                    01F0   433 10$:
                                    01F0   434          MODE    TO,20$,KRNL              ; GO TO KERNEL MODE TO ISSUE SERVICE
                                    0213   435          $EXPREG_S PAGCNT_ERG,RETADR_ERG, -
                                    0213   436                    ACMODE[R4],COND4_E[R5] ; ISSUE PRELIM S.S. TO GET SOME PAGES
                                    0234   437          MODE    FROM,20$                 ; BACK TO USER MODE
                0000000C'EF   D5    0235   438          TSTL    RETADR_ERG               ; DID EXPREG DO ANYTHING ?
                        6C    12    023B   439          BNEQ    30$                      ; YES -- THAT'S FINE
    00000000'EF   00000000'8F   D0    023D   440          MOVL    #SS$_NORMAL,EXPV         ; NO -- LOAD EXPECTED AND
             00000000'EF   50   D0    0248   441          MOVL    R0,RECV                  ; ... RECEIVED VALUES, THEN EXIT
                                    024F   442          ERR_EXIT LONG,<UNEXPECTED RETADR PAIR RETURNED BY NON-SUBJECT>,-
                                    024F   443                   < EXPREG>
                                    02A9   444 30$:
  00000030'EF   00000010'EF   D0    02A9   445          MOVL    RETADR_ERG+4,ENDADDR     ; GET ENDING ADDR OF AREA TO BE CREATED
  0000002C'EF   0000000C'EF   D0    02B4   446          MOVL    RETADR_ERG,STARTADDR     ; GET TENTATIVE START ADDR OF AREA
  00000034'EF   00000008'EF   D1    02BF   447          CMPL    PAGCNT_ERG,DISPL         ; WERE 5 PAGES ADDED TO DISPLACEMENT ?
                        1C    13    02CA   448          BEQL    45$                      ; NO -- LEAVE STARTADDR ALONE
             59   00000051'EF   D0    02CC   449          MOVL    PAGESIZE,R9              ; YES -- STARTADDR TO BE MODIFIED
             03 000001F7'EF45   E9    02D3   450          BLBC    COND4_E[R5],40$          ; IF PROGRAM REGION, LEAVE R9 ALONE
                      59   59   CE    02DB   451          MNEGL   R9,R9                    ; PROGRAM REGION -- NEGATE R9
                                    02DE   452 40$:
                      59   05   C4    02DE   453          MULL2   #5,R9                    ; GET LENGTH OF 5 PAGES
             0000002C'EF   59   C0    02E1   454          ADDL2   R9,STARTADDR             ; COMPUTE START ADDR OF AREA TO BE CREATED
                                    02E8   455 45$:
                                    02E8   456          MODE    TO,50$,KRNL              ; INTO KERNEL TO DO A STORE
  00000030'FF   00000000'EF   90    030B   457          MOVB    ONES,@ENDADDR            ; INDICATE WE WERE HERE
                                    0316   458          MODE    FROM,50$                 ; ... AND GET BACK TO USER
                   01   52   91    0317   459          CMPB    R2,#1                    ; 2ND CONDITION 1 ELEMENT ?
                        03   12    031A   460          BNEQU   73$                      ; NO -- CONTINUE
                      015F   31    031C   461          BRW     71$                      ; YES -- LEAVE EXPANDED REGION AS IS
                                    031F   462 73$:
                   02   52   91    031F   463          CMPB    R2,#2                    ; 3RD CONDITION 1 ELEMENT ?
                        03   12    0322   464          BNEQU   74$                      ; NO -- CONTINUE
                      00E8   31    0324   465          BRW     60$                      ; YES -- GO CONTRACT REGION
                                    0327   466 74$:
  00000014'EF   0000002C'EF   D0    0327   467          MOVL    STARTADDR,INADR_CVA      ; MUST BE 1ST COND 1 ELEMENT
  00000018'EF   00000030'EF   D0    0332   468          MOVL    ENDADDR,INADR_CVA+4      ; ... GET READY FOR DELTVA
                                    033D   469          MODE    TO,69$,KRNL              ; NEED KERNEL MODE
                                    0360   470          $DELTVA_S INADR=INADR_CVA, -
                                    0360   471                    ACMODE=ACMODE[R4]     ; DELETE EXPANDED AREA
                                    0376   472          MODE    FROM,69$                 ; BACK TO USER MODE
                                    0377   473          SS_CHECK NORMAL                  ; DID IT GO OK ?
                                    03A5   474          MODE    TO,70$,KRNL              ; INTO KERNEL FOR A PRELIM CRETVA
```

```
                                03C8   475          $CRETVA_S INADR=INADP_CVA, ACMODE=ACMODE[R4]
                                03DE   476          MODE    FROM,70$                ; USER MODE
                                03DF   477          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                          6F    11     040D   478          BRB     71$                     ; ... AND GO DETERMINE ORDER OF  INADR PAIR
                                040F   479 60$:
                                040F   480          MODE    TO,68$,KRNL             ; KERNEL MODE FOR CNTREG
                                0432   481          $CNTREG_S DISPL,,ACMODE[R4],COND4_E[R5]
                                044F   482                                          ; CONTRACT REQUESTED SIZE (NOT THE EXTRA 5)
                                044F   483          MODE    FROM,68$                ; ... AND GET BACK TO USER MODE
                                0450   484          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                                047E   485 71$:
00000030'EF   0000002C'EF       D1     047E   486          CMPL    STARTADDR,ENDADDR       ; FIND LARGER ADDRESS
                          10    1A     0489   487          BGTRU   72$                     ; STARTADDR LARGER
              5A  0000002C'EF   D0     048B   488          MOVL    STARTADDR,R10           ; SMALLER VALUE IN R10,
              5B  00000030'EF   D0     0492   489          MOVL    ENDADDR,R11             ; ... LARGER IN R11
                          0E    11     0499   490          BRB     75$                     ; CONTINUE
                                049B   491 72$:
              5A  00000030'EF   D0     049B   492          MOVL    ENDADDR,R10             ; SMALLER VALUE IN R10,
              5B  0000002C'EF   D0     04A2   493          MOVL    STARTADDR,R11           ; ... LARGER IN R11
                                04A9   494 75$:
                          53    95     04A9   495          TSTB    R3                      ; 1ST CONDITION 2 ELEMENT ?
                          10    13     04AB   496          BEQL    77$                     ; YES -- LOWER ADDRESS GOES FIRST
              0000001C'EF 5B    D0     04AD   497          MOVL    R11,INADR               ; NO -- HIGHER ADDRESS GOES FIRST
              00000020'EF 5A    D0     04B4   498          MOVL    R10,INADR+4             ; ........
                          0E    11     04BB   499          BRB     79$                     ; CONTINUE
                                04BD   500 77$:
              0000001C'EF 5A    D0     04BD   501          MOVL    R10,INADR               ; LOWER ADDRESS GOES FIRST
              00000020'EF 5B    D0     04C4   502          MOVL    R11,INADR+4             ; ........
                                04CB   503 79$:
                                04CB   504          MODE    TO,80$,KRNL             ; INTO KERNEL MODE
                                04EE   505 ;
                                04EE   506 ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                                04EE   507 ;
                                04EE   508          $CRETVA_S INADR,RETADR,ACMODE[R4]
                                0508   509          MODE    FROM,80$                ; BACK TO USER
        00000000'8F   50        D1     0509   510          CMPL    R0,#SS$_NORMAL          ; CODE RECEIVED = CODE EXPECTED ?
                          61    13     0510   511          BEQLU   81$                     ; YES -- MORE VERIFYING
00000000'EF   00000000'8F       D0     0512   512          MOVL    #SS$_NORMAL,EXPV        ; NO -- LOAD UP EXPECTED AND
        00000000'EF   50        D0     051D   513          MOVL    R0,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                                0524   514          ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM CRETVA>
                                0573   515 81$:
00000024'EF   0000001C'EF       D1     0573   516          CMPL    INADR,RETADR            ; DID CRETVA GIVE US THE 1ST PAGE ?
                          0F    12     057E   517          BNEQU   82$                     ; NO -- ERROR
00000028'EF   00000020'EF       D1     0580   518          CMPL    INADR+4,RETADR+4        ; HOW ABOUT THE WHOLE RANGE ?
                          02    12     058B   519          BNEQU   82$                     ; NO -- ERROR
                          67    11     058D   520          BRB     83$                     ; CONTINUE
                                058F   521 82$:
00000000'EF   0000001C'EF       7D     058F   522          MOVQ    INADR,EXPV              ; LOAD UP EXPECTED AND
00000000'EF   00000024'EF       7D     059A   523          MOVQ    RETADR,RECV             ; ... RECEIVED VALUES, THEN EXIT
                                05A5   524          ERR_EXIT QUAD,<UNEXPECTED VALUE FROM CRETVA FOR>,-
                                05A5   525                       < RETADR PAIR>
                                05F6   526 83$:
              03  0000015E'EF44    D1     05F6   527          CMPL    ACMODE[R4],#PSL$C_USER  ; USER MODE PAGES REQUESTED ?
                          52    13     05FE   528          BEQLU   90$                     ; YES -- GO ON
              00  0000015E'EF44    D1     0600   529          CMPL    ACMODE[R4],#PSL$C_KERNEL ; KERNEL MODE PAGES REQUESTED ?
                          25    13     0608   530          BEQLU   84$                     ; YES -- GO CHG TO KERNEL MODE
                                060A   531          MODE    TO,85$,EXEC             ; GET INTO EXEC MODE TO ACCESS NEW PAGES
```

```
                     23   11  062D   532          BRB     90$                         ; ... AND CONTINUE
                                 062F   533 84$:
                                 062F   534          MODE    TO,85$,KRNL                 ; GET INTO KERNEL MODE TO ACCESS NEW PAGES
                                 0652   535 90$:
                00000038'EF   94  0652   536          CLRB    NZERR                       ; INDICATE NO NON-ZERO ERROR
                                 0658   537 91$:
                         6A   95  0658   538          TSTB    (R10)                       ; DOES THIS PAGE HAVE A ZERO BYTE ?
                         09   12  065A   539          BNEQ    92$                         ; NO -- GO INDICATE ERROR
            6A   00000000'EF   90  065C   540          MOVB    ONES,(R10)                  ; STORE INTO PAGE -- NO ACCVIO EXPECTED
                         1A   11  0663   541          BRB     93$                         ; GO LOOP TO NEXT PAGE
                                 0665   542 92$:
 00000038'EF    00000000'EF   90  0665   543          MOVB    ONES,NZERR                  ; INDICATE A NON-ZERO ERROR WAS FOUND
                00000000'EF   94  0670   544          CLRB    EXPV                        ; LOAD UP EXPECTED AND
            00000000'EF   6A   90  0676   545          MOVB    (R10),RECV                  ; ... RECEIVED VALUES
                         0A   11  067D   546          BRB     100$                        ; GO GET USER MODE BACK
                                 067F   547 93$:
FFCF 5A    00000051'EF   5B   F1  067F   548          ACBL    R11,PAGESIZE,R10,91$        ; ADVANCE TO NEXT PAGE
                                 0689   549 100$:
         03   0000015E'EF44   D1  0689   550          CMPL    ACMODE[R4],#PSL$C_USER      ; USER MODE PAGES ?
                         01   13  0691   551          BEQLU   110$                        ; YES -- NO NEED TO CHG MODE
                                 0693   552          MODE    FROM,85$                    ; GET BACK TO USER MODE
                                 0694   553 110$:
                00000038'EF   95  0694   554          TSTB    NZERR                       ; NON-ZERO ERROR FOUND ?
                         4B   13  069A   555          BEQL    VERIFYX                     ; NO -- EVERYTHING VERIFIES
                                 069C   556          ERR_EXIT BYTE,<A PAGE IN THE CREATED AREA IS NON-ZERO>
                                 06E7   557 VERIFYX:
                                 06E7   558          RSB                                 ; RETURN TO CALLER
```

```
                     06E8   560          .SBTTL  VFY_CLEANUP
                     06E8   561 ;++
                     06E8   562 ; FUNCTIONAL DESCRIPTION:
                     06E8   563 ;
                     06E8   564 ;                 VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
                     06E8   565 ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
                     06E8   566 ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
                     06E8   567 ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
                     06E8   568 ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
                     06E8   569 ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
                     06E8   570 ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
                     06E8   571 ; POSSIBLY DISCOVERING A SECOND ERROR.
                     06E8   572 ;
                     06E8   573 ; CALLING SEQUENCE:
                     06E8   574 ;
                     06E8   575 ;       BSBW VFY_CLEANUP
                     06E8   576 ;
                     06E8   577 ; INPUT PARAMETERS:
                     06E8   578 ;
                     06E8   579 ;       NONE
                     06E8   580 ;
                     06E8   581 ; IMPLICIT INPUTS:
                     06E8   582 ;
                     06E8   583 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                     06E8   584 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                     06E8   585 ;       FOR X = 1,2,3,4,5 :
                     06E8   586 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
                     06E8   587 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
                     06E8   588 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
                     06E8   589 ;                 FOR CONDX_E.
                     06E8   590 ;
                     06E8   591 ; OUTPUT PARAMETERS:
                     06E8   592 ;
                     06E8   593 ;       NONE
                     06E8   594 ;
                     06E8   595 ; IMPLICIT OUTPUTS:
                     06E8   596 ;
                     06E8   597 ;       NONE
                     06E8   598 ;
                     06E8   599 ; COMPLETION CODES:
                     06E8   600 ;
                     06E8   601 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                     06E8   602 ;
                     06E8   603 ; SIDE EFFECTS:
                     06E8   604 ;
                     06E8   605 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                     06E8   606 ;       (VIA RSB) IF ERROR ENCOUNTERED.
                     06E8   607 ;
                     06E8   608 ;--
                     06E8   609
                     06E8   610
                     06E8   611
                     06E8   612 VFY_CLEANUP::
0000002C'EF    D5    06E8   613          TSTL    STARTADDR               ; DID REGION EVER GET ACQUIRED ?
          03   12    06EE   614          BNEQ    5$                      ; YES -- CONTINUE
        00F3   31    06F0   615          BRW     VFY_CLEANUPX            ; NO -- JUST GET OUT NOW
                     06F3   616 5$:
```

```
0000001C'EF   0000002C'EF   D0  06F3   617          MOVL    STARTADDR,INADR          ; GET READY FOR DELTVA
00000020'EF   00000030'EF   D0  06FE   618          MOVL    ENDADDR,INADR+4          ; ........
                            0709   619          MODE    TO,10$,KRNL              ; KERNEL MODE FOR DELETE
                            072C   620          $DELTVA_S INADR=INADR, ACMODE=ACMODE[R4]
                            0742   621                                           ; DELETE CREATED AREA
                            0742   622          MODE    FROM,10$                 ; BACK TO USER MODE
                            0743   623          SS_CHECK NORMAL                  ; CHECK FOR NORMAL RETURN
        01   000001F7'EF45  D1  0771   624          CMPL    COND4_E[R5],#1           ; WAS AREA IN "MIDDLE" OF A REGION ?
                       6B   1B  0779   625          BLEQU   VFY_CLEANUPX             ; NO -- TIME TO EXIT
                            077B   626          MODE    TO,20$,KRNL              ; YES -- KERNEL MODE TO ISSUE CNTREG
                            079E   627          $CNTREG_S #5,,ACMODE[R4],COND4_E[R5] ; GET RID OF THE EXTRA 5 PAGES
                            07B7   628          MODE    FROM,20$                 ; BACK TO USER MODE
                            07B8   629          SS_CHECK NORMAL                  ; CHECK FOR NORMAL RETURN
                            07E6   630 VFY_CLEANUPX:
                            07E6   631          RSB                              ; RETURN TO CALLER
                            07E7   632          .END
```

```
$$$$                    = 000006A6 R      04
$$$CHARS                = 00000026
$$$CHARS1               = 00000008
$$$CHARS2               = 0000000B
$$$CHARS3               = 0000000B
$$$CHARS4               = 00000000
$$$CHARS5               = 00000000
$$$COND_A               = 00000002
$$$STRINGS              = 00000001
$$$STRINGS2             = 00000005
$$T1                    = 00000000
$$T2                    = 00000004
ACMODE                    0000015E R      03
BYTE                    = 00000001   G
CFLAG                     ********     X    04
CHMRTN                    ********     X    04
CHM_CONT                  ********     X    04
COMP_SC                   ********     X    04
CONDT                     000000B7 RG    04
COND1_C                 = 00000000
COND1_CLEANUP             000000B8 RG    04
COND1_E                   000000BC R     03
COND1_H                   0000005D RG    03
COND1_T                   00000039 R     03
COND1_TAB                 0000005E R     03
COND2                     000000B9 RG    04
COND2_C                 = 00000000
COND2_CLEANUP             000000BA RG    04
COND2_E                   0000012F R     03
COND2_H                   000000D3 RG    03
COND2_T                   000000BC R     03
COND2_TAB                 000000D4 R     03
COND3                     000000BB RG    04
COND3_C                 = 00000004
COND3_CLEANUP             000000BC RG    04
COND3_E                   0000015E R     03
COND3_H                   00000136 RG    03
COND3_T                   0000012F R     03
COND3_TAB                 00000137 R     03
COND4                     000000BD RG    04
COND4_C                 = 00000000
COND4_CLEANUP             000000BE RG    04
COND4_E                   000001F7 R     03
COND4_H                   00000188 RG    03
COND4_T                   0000016E R     03
COND4_TAB                 00000189 R     03
COND5                     000000BF RG    04
COND5_C                 = 00000000
COND5_CLEANUP             000000C0 RG    04
COND5_E                   00000240 R     03
COND5_H                   00000212 RG    03
COND5_T                   00000207 R     03
COND5_TAB                 00000213 R     03
CTL$GL_PHD                ********     X    04
DESC                    = 00000010   G
DISPL                     00000034 R     03
EFLAG                     ********     X    04
ENDADDR                   00000030 R     03
EXPV                      ********     X    04
FAO_DESC                  ********     X    04
FAO_LEN                   ********     X    04
FORM_CONDS                000000C1 RG    04
FORM_CONDSX               000001B6 R     04
INADR                     0000001C R     03
INADR_CVA                 00000014 R     03
LONG                    = 00000004   G
MOD_MSG_CODE              ********     X    04
MOD_MSG_PRINT             ********     X    04
MSGT_INP_CTL              00000019 R     02
MSG3_ERR_CTL              00000039 RG    02
MSG_A                     ********     X    04
MSG_B                     ********     X    04
MSG_CTXT                  ********     X    04
MSG_DATA1                 ********     X    04
NOTARG                  = 00000000   G
NULL                    = 00000014   G
NZERR                     00000038 R     03
ONES                      ********     X    04
OUTPUT_MSG                ********     X    04
PAGCNT_ERG                00000008 R     03
PAGESIZE                  00000051 R     02
PCV                       ********     X    04
PHD$Q_PRIVMSK           = 00000000
PRIVMASK                  00000000 R     03
PRIV_ARGS               = 00000002
PROCESS_ERR               ********     X    04
PSL$C_EXEC              = 00000001
PSL$C_KERNEL            = 00000000
PSL$C_SUPER             = 00000002
PSL$C_USER              = 00000003
QUAD                    = 00000008   G
RECV                      ********     X    04
REST_REGS                 ********     X    04
RETADR                    00000024 R     03
RETADR_ERG                0000000C R     03
SAVE_REGS                 ********     X    04
SS$_NORMAL                ********     X    04
STARTADDR                 0000002C R     03
SUCCESS                   ********     X    04
SYS$CMEXEC                ********    GX    04
SYS$CMKRNL                ********    GX    04
SYS$CNTREG                ********    GX    04
SYS$CRETVA                ********    GX    04
SYS$DELTVA                ********    GX    04
SYS$EXPREG                ********    GX    04
SYS$FAO                   ********     X    04
SYS$SETPRN                ********    GX    04
SYS$SETPRV                ********    GX    04
TESTNUM                   ********     X    04
TEST_MOD_NAME             00000000 RG    02
TEST_MOD_NAME_D           00000009 R     02
TEST_MOD_SUCC             ********     X    04
TMD_ADDR                  ********     X    04
TM_CLEANUP                000000B3 RG    04
```

```
TM_SETUP                 00000000 RG    04
VERIFY                   000001B7 RG    04
VERIFYX                  000006E7 R     04
VFY_CLEANUP              000006E8 RG    04
VFY_CLEANUPX             000007E6 R     04
WORD                   = 00000002  G
WRITE_MSG2               ******** X     04
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
RODATA	00000055 (85.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	0000024C (588.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS72	000007E7 (2023.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	35	00:00:00.07	00:00:00.33
Command processing	133	00:00:00.65	00:00:02.72
Pass 1	269	00:00:08.49	00:00:16.59
Symbol table sort	0	00:00:00.56	00:00:01.15
Pass 2	130	00:00:02.31	00:00:04.55
Symbol table output	15	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	586	00:00:12.22	00:00:25.47

```
The working set limit was 1500 pages.
46744 bytes (92 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 350 non-local and 80 local symbols.
632 source lines were read in Pass 1, producing 27 object records in Pass 2.
38 pages of virtual memory were used to define 29 macros.
```

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SHRLIB]UETP.MLB;1	9
_255DUA28:[SYS.OBJ]LIB.MLB;1	1
_255DUA28:[SYSLIB]STARLET.MLB;2	16
TOTALS (all libraries)	26

```
677 GETS were required to define 26 macros.

There were no errors, warnings or information messages.
```

```
MACRO/LIS=LIS$:SATSSS72/OBJ=OBJ$:SATSSS72 MSRC$:SATSSS72/UPDATE=(ENH$:SATSSS72)+EXECML$/LIB+SHRLIB$:UETP/LIB
```

SATSSS56
LIS

SATSSS60
LIS

SATSSS71
LIS

SATSSS54
LIS

SATSSS73
LIS

SATSSS70
LIS

SATSSS72
LIS

SATSSS55
LIS

SATSSS53
LIS

SATSSS61
LIS