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

FILEID**SATSSS36

N 10

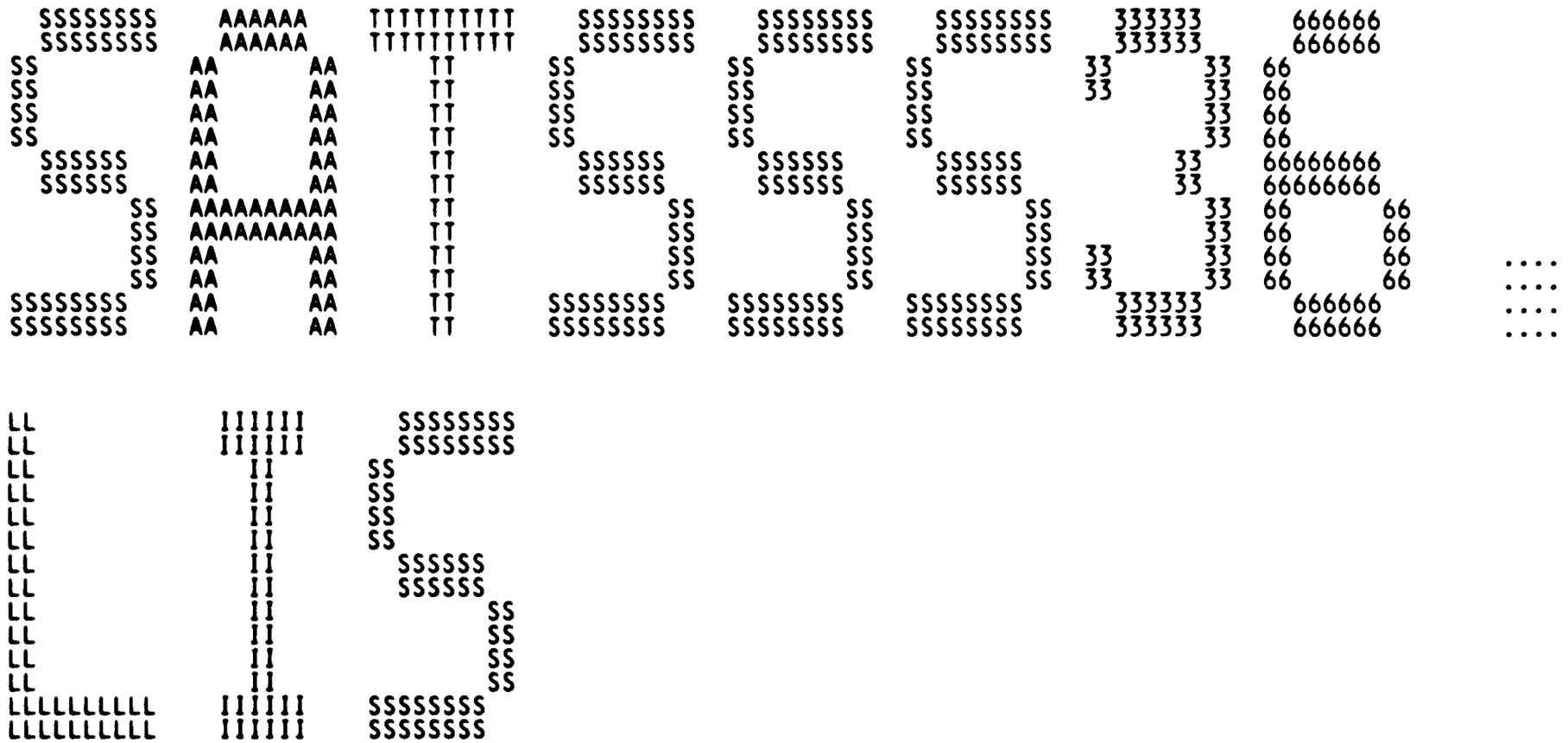

Table of contents

```
0000     1          .TITLE  SATSSS36 SATS SYSTEM SERVICE TESTS  $DELPRC (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 SATSSS36 TO TEST SUCCESSFUL
0000    36  ; OPERATION OF THE $DELPRC SYSTEM SERVICE. THE SERVICE IS INVOKED
0000    37  ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38  ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39  ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40  ; CHECKING FOR AN SS$_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: MAR, 1978
0000    47  ;
0000    48  ; MODIFIED BY:
0000    49  ;
0000    50  ;       V03-001 LDJ0001         Larry D. Jones,         23-Jun-1983
0000    51  ;                Set quotas to ones to force the use of the sysboot minimum
0000    52  ;                values so that they could be tested.
0000    53  ;--
```

```
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          $PQLDEF                         ; PROCESS QUOTA CODES
0000     62          $PCBDEF                         ; PCB LABELS
0000     63          $MSGDEF                         ; MAILBOX MSG IDENTIFIERS
0000     64          $LOGDEF                         ; LOGICAL NAME TABLE TYPE SYMBOLS
0000     65          $DIBDEF                         ; DEVICE INFO BLOCK OFFSETS
0000     66 ;
0000     67 ; MACROS:
0000     68 ;
0000     69 ;
0000     70 ; EQUATED SYMBOLS:
0000     71 ;
0000     72 ;
0000     73 ; OWN STORAGE:
0000     74 ;
```

```
                    00000000     76          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
                        0000     77 TEST_MOD_NAME:: STRING  C,<SATSSS36>     ; TEST MODULE NAME
                        0009     78 TEST_MOD_NAME_D: STRING I,<SATSSS36>     ; TEST MODULE NAME DESCRIPTOR
                        0019     79 MSG1_INP_CTL:   STRING  I,<  SSDPC!4ZW: CONDITIONS:>
                        0039     80                                          ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
                        0039     81 MSG3_ERR_CTL::  STRING  I,< *SSDPC!4ZW: !AS>
                        0051     82                                          ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
            FFFFFFFF DC3CBA00    83 ONE_MIN:        .LONG   -10*1000*1000*60,-1 ; ONE MINUTE (WAKE-UP DELTA)
                        0059     84 CREATED_PRN:    STRING  I,<SATSSS36_CRE> ; PROCESS & MBX NAME FOR CREATED PROCESS
                        006D     85 IMAGNAM:        STRING  I,<SYSTST$RES:SATSUT14.EXE> ; IMAGE NAME FOR CREATED PROC
                        008C     86 LOGNAM_PID:     STRING  I,<SYSTST$PID>   ; LOG NAME OF CREATING PID
                    00000004     87 EQUIV_PID:      .LONG   4                ; EQUIV NAME STRING DESCRIPTOR
                    00000114'    88                 .ADDRESS CREATING_PID    ; ... OF CREATING PID
                        00A6     89 ;QUOTALIST:     $QUOTA  CPULM,0          ; INFINITE CPU
                        00A6     90 ;               $QUOTA  BYTLM,1          ; Force minimums to be used
                        00A6     91 ;               $QUOTA  FILLM,1
                        00A6     92 ;               $QUOTA  PGFLQUOTA,1
                        00A6     93 ;               $QUOTA  PRCLM,1
                        00A6     94 ;               $QUOTA  TQELM,1
                        00A6     95 ;               $QUOTA  LISTEND          ; DEFINES END OF LIST
```

```
                00000000     97          .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008  0000     98 PRIVMASK:        .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000C  0008     99 MBXCHAN:         .BLKL   1               ; CHAN NO. FOR MAILBOX FOR CREATED PROCESS
          000C    100 MBXCHANINFO:                             ; CHANNEL INFO RETURNED BY GETCHN
00000074  000C    101                 .LONG   DIB$K_LENGTH
00000014'  0010   102                 .ADDRESS .+4
00000088  0014    103                 .BLKB   DIB$K_LENGTH
0000008C  0088    104 MBXUNIT:        .BLKL   1               ; SAVE AREA FOR MAILBOX UNIT NUMBER
          008C    105 MBXBUFF:        STRING  0,120           ; MAILBOX BUFFER FOR CREATED PROCESS
00000110  010C    106 DEST_PIDADR:    .BLKL   1               ; DESTINATION PID ADDR, WRITTEN BY SUBJ S.S.
00000114  0110    107 ZEROPID:        .BLKL   1               ; PID OF ZEROES
00000000  0114    108 CREATING_PID:   .LONG   0               ; PID OF CREATING PROCESS
0000011C  0118    109 CREATED_PID:    .BLKL   1               ; PID OF CREATED PROCESS
```

```
                     011C    111          .SBTTL  CONDITION TABLES
                     011C    112 ;
                     011C    113 ;        ***** CONDITION TABLES FOR DELPRC SYSTEM SERVICE *****
                     011C    114 ;
                     011C    115          COND    1,NOTARG,<PID ADDRESS>,-
                     011C    116                    <NOT SPECIFIED>,-
                     011C    117                    <SPECIFIED, NON-ZERO>,-
                     011C    118                    <SPECIFIED, ZERO>,-
                     011C    119
00000000'            0167    120                      .ADDRESS    0
00000118'            016B    121                      .ADDRESS    CREATED_PID
00000110'            016F    122                      .ADDRESS    ZEROPID
                     0173    123 ;
                     0173    124          COND    2,NOTARG,<PROCESS NAME ADDRESS>,-
                     0173    125                    <SPECIFIED>,-
                     0173    126                    <NOT SPECIFIED>,-
                     0173    127
00000059'            01A9    128                      .ADDRESS    CREATED_PRN
00000000'            01AD    129                      .ADDRESS    0
                     01B1    130 ;
                     01B1    131          COND    3,NOTARG,<CREATED PROCESS TYPE>,-
                     01B1    132                    <SUBPROCESS>,-
                     01B1    133                    <DETACHED, DIFFERENT GROUP>,-
                     01B1    134                    <DETACHED, SAME GROUP, SAME MEMBER>,-
                     01B1    135                    <DETACHED, SAME GROUP, DIFFERENT MEMBER>,-
                     01B1    136
00000000             0245    137                      .LONG       0           ; PSEUDO-UIC
0000024D             0249    138                      .BLKL       1           ; UIC
00000251             024D    139                      .BLKL       1           ; UIC
00000255             0251    140                      .BLKL       1           ; UIC
                     0255    141 ;
                     0255    142          COND    4,NULL
                     0256    143
                     0256    144          COND    5,NULL
                     0257    145
00000000             0256    146          .PSECT  SATSSS36,RD,WRT,EXE
```

```
                     0000   148          .SBTTL  TM_SETUP, TM_CLEANUP
                     0000   149 ;++
                     0000   150 ; FUNCTIONAL DESCRIPTION:
                     0000   151 ;
                     0000   152 ;               TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                     0000   153 ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                     0000   154 ; TEST MODULE EXECUTION.
                     0000   155 ;
                     0000   156 ; CALLING SEQUENCE:
                     0000   157 ;
                     0000   158 ;       BSBW TM_SETUP   BSBW TM_CLEANUP
                     0000   159 ;
                     0000   160 ; INPUT PARAMETERS:
                     0000   161 ;
                     0000   162 ;       NONE
                     0000   163 ;
                     0000   164 ; IMPLICIT INPUTS:
                     0000   165 ;
                     0000   166 ;       NONE
                     0000   167 ;
                     0000   168 ; OUTPUT PARAMETERS:
                     0000   169 ;
                     0000   170 ;       NONE
                     0000   171 ;
                     0000   172 ; IMPLICIT OUTPUTS:
                     0000   173 ;
                     0000   174 ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                     0000   175 ;                  ALL PRIVILEGES ACQUIRED.
                     0000   176 ;
                     0000   177 ; COMPLETION CODES:
                     0000   178 ;
                     0000   179 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                     0000   180 ;
                     0000   181 ; SIDE EFFECTS:
                     0000   182 ;
                     0000   183 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                     0000   184 ;       (VIA RSB) IF ERROR ENCOUNTERED.
                     0000   185 ;
                     0000   186 ;--
                     0000   187
                     0000   188
                     0000   189
                     0000   190 TM_SETUP::
                       52 D4 0000   191          CLRL    R2                          ; INITIALIZE
                       53 D4 0002   192          CLRL    R3                          ; .. CONDITION
                       54 D4 0004   193          CLRL    R4                          ; .... TABLE
                       55 D4 0006   194          CLRL    R5                          ; ...... INDEX
                       56 D4 0008   195          CLRL    R6                          ; ........ REGISTERS
                   FFF3' 30 000A   196          BSBW    MOD_MSG_PRINT               ; PRINT TEST MODULE BEGIN MSG
00000000'EF 00000000'EF DE 000D   197          MOVAL   TEST_MOD_SUCC,TMD_ADDR      ; ASSUME END MSG WILL SHOW SUCCESS
   03 00 00000000'8F F0 0018   198          INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
         00000000'EF      0020
                     0025   199          MODE    TO,5$,KRNL                  ; KERNEL MODE TO ACCESS PHD
         59 00000000'9F D0 0048   200          MOVL    @#CTL$GL_PHD,R9             ; GET PROCESS HEADER ADDRESS
         00000000'EF 69 DE 004F   201          MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                     0056   202          MODE    FROM,5$ ; BACK TO USER MODE
                     0057   203          PRIV    ADD,ALL                     ; GET ALL PRIVILEGES
```

```
                                0077   204          $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                                0084   205          SS_CHECK  NORMAL                ; CHECK STATUS CODE RETURNED FROM SETPRN
                                00B2   206          $WAKE_S PIDADR=CREATING_PID     ; GET MY PID
                                00C1   207          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                                00EF   208          $HIBER_S                        ; UNDO ABOVE WAKE
                                00F6   209          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                                0124   210          $CRELOG_S TBLFLG=#LOG$C_SYSTEM, - ; GET MY PID INTO LOG NAME TABLE
                                0124   211                    LOGNAM=LOGNAM_PID, -  ; ... FOR USE BY CREATED PROCESS
                                0124   212                    EQLNAM=EQUIV_PID
                     2E 50   E8 013B   213          BLBS    R0,10$                  ; IF SUCCESSFUL, CONTINUE
                                013E   214          SS_CHECK NORMAL                 ; USE SS_CHECK TO TERMINATE MODULE
                                016C   215 10$:
                                016C   216 ;
                                016C   217 ; THE FOLLOWING CODE ESTABLISHES UIC'S IN THE CONDITION 3 TABLE
                                016C   218 ;
                                016C   219          MODE    TO,20$,KRNL             ; KERNEL MODE TO ACCESS PCB
          59   00000000'9F    D0 018F   220          MOVL    @#SCH$GL_CURPCB,R9      ; GET CURRENT PCB ADDRESS
               59   00BC C9   D0 0196   221          MOVL    PCB$L_UIC(R9),R9        ; PICK UP UIC FROM PCB
                                019B   222          MODE    FROM,20$                ; ... AND GET BACK TO USER MODE
                                019C   223 ; R9 NOW CONTAINS "MY" UIC
                   5A   01   9A 019C   224          MOVZBL  #1,R10                  ; GET COND3 TABLE INDEX NUMBER INTO A REG
          59   00010000 8F   C1 019F   225          ADDL3   #^X10000,R9,COND3_E[R10] ; PUT DIFF GROUP UIC INTO 2ND TABLE ELT
             00000245'EF4A            01A6
                      5A   D6 01AC   226          INCL    R10                     ; POINT TO 3RD COND3 TABLE ELEMENT
     00000245'EF4A    59   D0 01AE   227          MOVL    R9,COND3_E[R10]         ; PUT MY UIC INTO TABLE
                      5A   D6 01B6   228          INCL    R10                     ; POINT TO 4TH COND3 TABLE ELEMENT
00000245'EF4A    59   01   C1 01B8   229          ADDL3   #1,R9,COND3_E[R10]      ; PUT DIFF MEMBER UIC INTO THE TABLE
                                01C1   230          $CREMBX_S CHAN=MBXCHAN, LOGNAM=CREATED_PRN, -
                                01C1   231                    MAXMSG=#120, PROMSK=#0, BUFQUO=#240
                                01E6   232                                          ; GET MAILBOX FOR PROCESS
                                01E6   233          SS_CHECK NORMAL                 ; CHECK NORMAL COMPLETION
                                0214   234          $GETCHN_S CHAN=MBXCHAN, PRIBUF=MBXCHANINFO
                                022E   235                                          ; GET CHAN INFO (UNIT NUMBER)
                                022E   236          SS_CHECK NORMAL                 ; CHECK NORMAL COMPLETION
00000088'EF    00000020'EF   3C 025C   237          MOVZWL  MBXCHANINFO+8+DIB$W_UNIT,MBXUNIT
                                0267   238                                          ; SAVE MAILBOX UNIT NUMBER
                             05 0267   239          RSB                             ; RETURN TO MAIN ROUTINE
                                0268   240 TM_CLEANUP::
                                0268   241          $DELMBX_S  MBXCHAN              ; DELETE TERMINATION MAILBOX
                                0276   242          $DELLOG_S TBLFLG=#LOG$C_SYSTEM, - ; DELETE LOG NAME ACQUIRED ABOVE
                                0276   243                    LOGNAM=LOGNAM_PID
                         FD76'  30 0287   244          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE END MSG
                             05 028A   245          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                     028B    247          .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
                     028B    248 ;++
                     028B    249 ; FUNCTIONAL DESCRIPTION:
                     028B    250 ;
                     028B    251 ;               CONDX AND CONDX_CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
                     028B    252 ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
                     028B    253 ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
                     028B    254 ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
                     028B    255 ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
                     028B    256 ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
                     028B    257 ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
                     028B    258 ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
                     028B    259 ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
                     028B    260 ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
                     028B    261 ;
                     028B    262 ; CALLING SEQUENCE:
                     028B    263 ;
                     028B    264 ;       BSBW CONDX   BSBW CONDX_CLEANUP
                     028B    265 ;         WHERE X = 1,2,3,4,5
                     028B    266 ;
                     028B    267 ; INPUT PARAMETERS:
                     028B    268 ;
                     028B    269 ;       CONFLICT = 0
                     028B    270 ;
                     028B    271 ; IMPLICIT INPUTS:
                     028B    272 ;
                     028B    273 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                     028B    274 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                     028B    275 ;
                     028B    276 ; OUTPUT PARAMETERS:
                     028B    277 ;
                     028B    278 ;       CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
                     028B    279 ;
                     028B    280 ; IMPLICIT OUTPUTS:
                     028B    281 ;
                     028B    282 ;       R2,3,4,5,6 PRESERVED
                     028B    283 ;
                     028B    284 ; COMPLETION CODES:
                     028B    285 ;
                     028B    286 ;       NONE
                     028B    287 ;
                     028B    288 ; SIDE EFFECTS:
                     028B    289 ;
                     028B    290 ;       NONE
                     028B    291 ;
                     028B    292 ;--
                     028B    293
                     028B    294
                     028B    295
                     028B    296 COND1::
                  05 028B    297          RSB                                ; RETURN TO MAIN ROUTINE
                     028C    298 COND1_CLEANUP::
                  05 028C    299          RSB                                ; RETURN TO MAIN ROUTINE
                     028D    300 COND2::
00000167'EF42 00000110'8F    D1 028D    301          CMPL    #ZEROPID,COND1_E[R2]    ; PID SPECIFIED AS 0 ??
                   14        12 0299    302          BNEQU   COND2X                  ; NO -- NO CONFLICT
              000001A9'EF43  D5 029B    303          TSTL    COND2_E[R3]             ; YES -- IS THERE A PROCESS NAME ??
```

```
                     0B  12  02A2   304          BNEQU   COND2X                          ; YES -- NO CONFLICT
     00000000'EF   00000000'EF  90  02A4   305          MOVB    ONES,CONFLICT           ; NO -- INDICATE CONFLICT BECAUSE THIS TYPE
                                    02AF   306                                          ; ... OF DELPRC WOULD DELETE CREATING PROC
                                    02AF   307 COND2X:
                                 05 02AF   308          RSB                             ; RETURN TO MAIN ROUTINE
                                    02B0   309 COND2_CLEANUP::
                                 05 02B0   310          RSB                             ; RETURN TO MAIN ROUTINE
                                    02B1   311 COND3::
   00000167'EF42   00000118'8F  D1  02B1   312          CMPL    #CREATED_PID,COND1_E[R2] ; NON-ZERO PID SPECIFIED ?
                           19   13  02BD   313          BEQLU   COND3X                  ; YES -- NO CONFLICT
                 000001A9'EF43  D5  02BF   314          TSTL    COND2_E[R3]             ; IS PROCESS NAME SPECIFIED ?
                           10   13  02C6   315          BEQL    COND3X                  ; NO -- NO CONFLICT
                                    02C8   316 ;
                                    02C8   317 ; NOTE -- AT THIS POINT, PROCESS WILL BE REFERENCED BY PROCESS NAME.
                                    02C8   318 ;
                        01   54  D1 02C8   319          CMPL    R4,#1                   ; DOES CONDITION 3 SPECIFY DIFFERENT GROUP ?
                           0B   12  02CB   320          BNEQ    COND3X                  ; NO -- NO CONFLICT
     00000000'EF   00000000'EF  90  02CD   321          MOVB    ONES,CONFLICT           ; YES -- PROCESS NAME FOR DIFF GROUP IS CONF
                                    02D8   322 COND3X:
                                 05 02D8   323          RSB                             ; RETURN TO MAIN ROUTINE
                                    02D9   324 COND3_CLEANUP::
                                 05 02D9   325          RSB                             ; RETURN TO MAIN ROUTINE
                                    02DA   326 COND4::
                                 05 02DA   327          RSB                             ; RETURN TO MAIN ROUTINE
                                    02DB   328 COND4_CLEANUP::
                                 05 02DB   329          RSB                             ; RETURN TO MAIN ROUTINE
                                    02DC   330 COND5::
                                 05 02DC   331          RSB                             ; RETURN TO MAIN ROUTINE
                                    02DD   332 COND5_CLEANUP::
                                 05 02DD   333          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                                        .SBTTL  FORM_CONDS
                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;
                                ;               FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                                ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       BSBW FORM_CONDS
                                ;
                                ; INPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT INPUTS:
                                ;
                                ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                                ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                                ;       FOR X = 1,2,3,4,5 :
                                ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                                ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                                ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                                ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                                ;
                                ; OUTPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT OUTPUTS:
                                ;
                                ;       NONE
                                ;
                                ; COMPLETION CODES:
                                ;
                                ;       NONE
                                ;
                                ; SIDE EFFECTS:
                                ;
                                ;       NONE
                                ;
                                ;--
                                
                                
                                FORM_CONDS::
                                        $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                                                                        ; FORMAT CONDITIONS HEADER MSG
                      FD00'  30         BSBW    OUTPUT_MSG            ; ... AND PRINT IT
                  14    00   91         CMPB    #COND1_C,#NULL        ; IS CONDITION 1 NULL ?
                        03   12         BNEQU   10$                   ; NO -- CONTINUE
                     00BF    31         BRW     FORM_CONDSX           ; YES -- SUBROUTINE IS FINISHED
                                10$:
  00000000'EF    0000011C'EF    DE         MOVAL   COND1_T,MSG_A         ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF   00000129'EF42    D0         MOVL    COND1_TAB[R2],MSG_B   ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
            00000000'EF    00   90         MOVB    #COND1_C,MSG_CTXT     ; SAVE CONDITION 1 CONTEXT FOR FAO
                                        MOV_VAL COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                      FCD7'  30  0326    392          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 1 MSG
                  14    00   91  0329    393          CMPB    #COND2_C,#NULL              ; IS CONDITION 2 NULL ?
                        03   12  032C    394          BNEQU   20$                         ; NO -- CONTINUE
                      0096   31  032E    395          BRW     FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
                             0331    396 20$:
  00000000'EF   00000173'EF   DE  0331    397          MOVAL   COND2_T,MSG_A               ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF   00000189'EF43   D0  033C    398          MOVL    COND2_TAB[R3],MSG_B         ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
          00000000'EF   00    90  0348    399          MOVB    #COND2_C,MSG_CTXT           ; SAVE CONDITION 2 CONTEXT FOR FAO
                             034F    400          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                      FCAE'  30  034F    401          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 2 MSG
                  14    00   91  0352    402          CMPB    #COND3_C,#NULL              ; IS CONDITION 3 NULL ?
                        03   12  0355    403          BNEQU   30$                         ; NO -- CONTINUE
                      006D   31  0357    404          BRW     FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
                             035A    405 30$:
  00000000'EF   000001B1'EF   DE  035A    406          MOVAL   COND3_T,MSG_A               ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF   000001C7'EF44   D0  0365    407          MOVL    COND3_TAB[R4],MSG_B         ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
          00000000'EF   00    90  0371    408          MOVB    #COND3_C,MSG_CTXT           ; SAVE CONDITION 3 CONTEXT FOR FAO
                             0378    409          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                      FC85'  30  0378    410          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 3 MSG
                  14    14   91  037B    411          CMPB    #COND4_C,#NULL              ; IS CONDITION 4 NULL ?
                        47   13  037E    412          BEQLU   FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   00000255'EF   DE  0380    413          MOVAL   COND4_T,MSG_A               ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF   00000255'EF45   D0  038B    414          MOVL    COND4_TAB[R5],MSG_B         ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
          00000000'EF   14    90  0397    415          MOVB    #COND4_C,MSG_CTXT           ; SAVE CONDITION 4 CONTEXT FOR FAO
                             039E    416          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                      FC5F'  30  039E    417          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 4 MSG
                  14    14   91  03A1    418          CMPB    #COND5_C,#NULL              ; IS CONDITION 5 NULL ?
                        21   13  03A4    419          BEQLU   FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   00000256'EF   DE  03A6    420          MOVAL   COND5_T,MSG_A               ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF   00000256'EF46   D0  03B1    421          MOVL    COND5_TAB[R6],MSG_B         ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
          00000000'EF   14    90  03BD    422          MOVB    #COND5_C,MSG_CTXT           ; SAVE CONDITION 5 CONTEXT FOR FAO
                             03C4    423          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                      FC39'  30  03C4    424          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 5 MSG
                             03C7    425 FORM_CONDSX:
                        05   03C7    426          RSB                                 ; RETURN TO CALLER
```

```
03C8    428          .SBTTL  VERIFY
03C8    429 ;++
03C8    430 ; FUNCTIONAL DESCRIPTION:
03C8    431 ;
03C8    432 ;               VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
03C8    433 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
03C8    434 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
03C8    435 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
03C8    436 ; ($DELPRC). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
03C8    437 ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
03C8    438 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
03C8    439 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
03C8    440 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
03C8    441 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
03C8    442 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
03C8    443 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
03C8    444 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
03C8    445 ;
03C8    446 ; CALLING SEQUENCE:
03C8    447 ;
03C8    448 ;       BSBW VERIFY
03C8    449 ;
03C8    450 ; INPUT PARAMETERS:
03C8    451 ;
03C8    452 ;       NONE
03C8    453 ;
03C8    454 ; IMPLICIT INPUTS:
03C8    455 ;
03C8    456 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
03C8    457 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
03C8    458 ;       FOR X = 1,2,3,4,5 :
03C8    459 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
03C8    460 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
03C8    461 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
03C8    462 ;                 FOR CONDX_E.
03C8    463 ;
03C8    464 ; OUTPUT PARAMETERS:
03C8    465 ;
03C8    466 ;       NONE
03C8    467 ;
03C8    468 ; IMPLICIT OUTPUTS:
03C8    469 ;
03C8    470 ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
03C8    471 ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
03C8    472 ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
03C8    473 ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
03C8    474 ;       ERRORS.
03C8    475 ;
03C8    476 ; COMPLETION CODES:
03C8    477 ;
03C8    478 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
03C8    479 ;
03C8    480 ; SIDE EFFECTS:
03C8    481 ;
03C8    482 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
03C8    483 ;       (VIA RSB) IF ERROR ENCOUNTERED.
03C8    484 ;
```

```
                     03C8   485 ;--
                     03C8   486
                     03C8   487
                     03C8   488
                     03C8   489 VERIFY::
00000000'EF    95    03C8   490         TSTB    CFLAG                   ; SHOULD CONDITIONS BE PRINTED ?
          03   13    03CE   491         BEQL    5$                      ; NO -- CONTINUE
        FF0B   30    03D0   492         BSBW    FORM_CONDS              ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                     03D3   493 5$:
00000110'EF    D4    03D3   494         CLRL    ZEROPID                 ; CLEAR ZERO PID
                     03D9   495         $CREPRC_S PIDADR=CREATED_PID, PRCNAM=CREATED_PRN, -
                     03D9   496                   UIC=COND3_E[R4], IMAGE=IMAGNAM, -
                     03D9   497                   MBXUNT=MBXUNIT;, QUOTA=QUOTALIST
                     0410   498                                           ; CREATE A PROCESS TO BE DELPRC'D
                     0410   499         SS_CHECK NORMAL                   ; ... AND MAKE SURE IT CREATED OK
                     043E   500         $SCHDWK_S DAYTIM=ONE_MIN          ; WAKE SELF IN 1 MIN IF CREATED PROC DOESN'T
                     0451   501         SS_CHECK NORMAL                   ; CHECK FOR NORMAL RETURN
                     047F   502         $HIBER_S                          ; SLEEP UNTIL CREATED PROC IS FULLY CREATED
                     0486   503         SS_CHECK NORMAL                   ; EXPECT NORMAL RETURN
                     04B4   504         $CANWAK_S                         ; GET RID OF SCHEDULED WAKE-UP
                     04BF   505         SS_CHECK NORMAL                   ; CHECK FOR NORMAL STATUS RETURN
                     04ED   506 ;
                     04ED   507 ; SCHEDULED WAKE-UP WILL ONLY BE EFFECTED IF CREATED PROCESS DOES
                     04ED   508 ; NOT GET FULLY CREATED. IN THIS CASE, THE SUBJECT SYSTEM SERVICE
                     04ED   509 ; BELOW WILL FAIL WITH AN APPROPRIATE ERROR CONDITION.
                     04ED   510 ;
                     04ED   511 ;
                     04ED   512 ; THE FOLLOWING CODE LOOKS FOR THE SPECIAL CASE OF NO PID SPECIFIED
                     04ED   513 ; AND NO PROCESS NAME SPECIFIED IN CONDITION TABLES. IF THIS CASE
                     04ED   514 ; IS PRESENT, DELPRC IS NOT ISSUED HERE, BUT, INSTEAD, A $WAKE IS
                     04ED   515 ; ISSUED FOR THE CREATED PROCESS, WHICH, IN TURN, ISSUES A $DELPRC
                     04ED   516 ; TO DELETE ITSELF. FOR ALL OTHER CASES, THE CREATED PROCESS
                     04ED   517 ; IS DELETED BY A $DELPRC ISSUED HERE IN THIS PROCESS.
                     04ED   518 ;
     00000167'EF42   D5    04ED   519         TSTL    COND1_E[R2]             ; IS PIDADR SPECIFIED ??
               49    12    04F4   520         BNEQU   10$                     ; YES -- NO SPECIAL CASE -- CONTINUE
     000001A9'EF43   D5    04F6   521         TSTL    COND2_E[R3]             ; NO -- HOW ABOUT PROCESS NAME ??
               40    12    04FD   522         BNEQU   10$                     ; IT EXISTS -- A NORMAL CASE
                     04FF   523         $WAKE_S PIDADR=CREATED_PID      ; NO PIDADR OR PIDADR SPECIFIED
                     050E   524                                         ; WAKE CREATED PROCESS TO DELETE ITSELF
                     050E   525         SS_CHECK NORMAL                 ; CHECK FOR NORMAL STATUS RETURN
              00FD   31    053C   526         BRW     20$                     ; ... AND GO WAIT FOR ITS MAIL
                     053F   527 10$:
                     053F   528 ;
                     053F   529 ; SET UP TO ISSUE SUBJECT $DELPRC IN THIS PROCESS
                     053F   530 ;
0000010C'EF  00000167'EF42   D0    053F   531         MOVL    COND1_E[R2],DEST_PIDADR ; GET PID ADDRESS OUT OF TABLE
          59 000001A9'EF43   D0    054B   532         MOVL    COND2_E[R3],R9          ; PRCNAM ADDRESS INTO REG FOR INDIRECT REF
                     0553   533 ;
                     0553   534 ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                     0553   535 ;
                     0553   536         $DELPRC_S PIDADR=@DEST_PIDADR, PRCNAM=(R9)
          00000000'8F   50    D1    0562   537         CMPL    R0,#SS$_NORMAL          ; CODE RECEIVED = CODE EXPECTED ?
                    61   13    0569   538         BEQLU   15$                     ; YES -- CONTINUE
00000000'EF  00000000'8F   D0    056B   539         MOVL    #SS$_NORMAL,EXPV        ; NO -- LOAD UP EXPECTED AND ...
          00000000'EF   50    D0    0576   540         MOVL    R0,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                     057D   541         ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM DELPRC>
```

```
                                05CC    542 15$:
                0000010C'EF     D5  05CC    543         TSTL    DEST_PIDADR             ; PID RETURNED BY DELPRC ?
                           68   13  05D2    544         BEQL    20$                     ; NO -- KEEP GOING
    0000010C'FF 00000118'EF     D1  05D4    545         CMPL    CREATED_PID,@DEST_PIDADR ; YES -- IS IT THE CORRECT ONE ?
                           5B   13  05DF    546         BEQL    20$                     ; YES -- CONTINUE
    00000000'EF 00000118'EF     D0  05E1    547         MOVL    CREATED_PID,EXPV        ; NO --LOAD UP EXPECTED AND ...
    00000000'EF 0000010C'FF     D0  05EC    548         MOVL    @DEST_PIDADR,RECV       ; ... RECEIVED VALUES, THEN EXIT
                                05F7    549         ERR_EXIT LONG,<INCORRECT PID RETURNED BY DELPRC>
                                063C    550 20$:
                                063C    551 ;
                                063C    552 ; CREATED PROCESS HAS BEEN DELPRC'D (BY THIS PROCESS OR BY ITSELF)
                                063C    553 ;
                                063C    554         $QIOW_S CHAN=MBXCHAN, FUNC=#IO$_READVBLK, -
                                063C    555                 P1=MBXBUFF+8, P2=MBXBUFF
                                0665    556                                         ; WAIT FOR CREATED PROCESS TO SEND MAIL
                                0665    557         SS_CHECK NORMAL                 ; CHECK FOR NORMAL STATUS CODE
          03    00000094'EF     B1  0693    558         CMPW    MBXBUFF+8,#MSG$_DELPROC ; DOES MAILBOX HAVE TERMINATION MSG ??
                           57   13  069A    559         BEQLU   30$                     ; YES -- AS EXPECTED
          00    000'EF    03    B0  069C    560         MOVW    #MSG$_DELPROC,EXPV      ; NO -- LOAD UP EXPECTED AND
    00000000'EF 00000094'EF     B0  06A3    561         MOVW    MBXBUFF+8,RECV          ; ... RECEIVED VALUES, THEN EXIT
                                06AE    562         ERR_EXIT WORD,<INCORRECT MAILBOX MSG IDENTIFIER>
                                06F3    563 30$:
    00000118'EF 0000009C'EF     D1  06F3    564         CMPL    MBXBUFF+16,CREATED_PID  ; DOES MAILBOX MSG HAVE CORRECT PID ??
                           5C   13  06FE    565         BEQLU   40$                     ; YES -- AS EXPECTED
    00000000'EF 00000118'EF     D0  0700    566         MOVL    CREATED_PID,EXPV        ; NO -- LOAD UP EXPECTED AND
    00000000'EF 0000009C'EF     D0  070B    567         MOVL    MBXBUFF+16,RECV         ; ... RECEIVED VALUES, THEN EXIT
                                0716    568         ERR_EXIT LONG,<INCORRECT PID RETURNED IN MAILBOX>
                                075C    569 40$:
                00000098'EF     D5  075C    570         TSTL    MBXBUFF+12              ; DOES MAILBOX MSG INDICATE 0 STATUS CODE ??
                           5F   13  0762    571         BEQLU   VERIFYX                 ; YES -- ALL IS OK
                00000000'EF     D4  0764    572         CLRL    EXPV                    ; NO -- LOAD UP EXPECTED AND
    00000000'EF 00000098'EF     D0  076A    573         MOVL    MBXBUFF+12,RECV         ; ... RECEIVED VALUES, THEN EXIT
                                0775    574         ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED IN MAILBOX>
                                07C3    575 VERIFYX:
                           05   07C3    576         RSB                             ; RETURN TO CALLER
```

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


                VFY_CLEANUP::
                        $DELPRC_S PRCNAM=CREATED_PRN    ; DELETE CREATED PROCESS (IF STILL HERE)
05              RSB                                     ; RETURN TO CALLER
                .END
```

```
$$$$                    = 0000077F R     04
$$$CHARS                = 00000029
$$$CHARS1               = 0000000A
$$$CHARS2               = 00000019
$$$CHARS3               = 00000021
$$$CHARS4               = 00000026
$$$CHARS5               = 00000000
$$$COND_A               = 00000003
$$$STRINGS              = 00000001
$$$STRINGS2             = 00000005
$$T1                    = 00000001
$$T2                    = 00000004
BYTE                    = 00000001  G
CFLAG                     ********    X   04
CHMRTN                    ********    X   04
CHM_CONT                  ********    X   04
COMP_SC                   ********    X   04
COND1                     0000028B RG     04
COND1_C                 = 00000000
COND1_CLEANUP             0000028C RG     04
COND1_E                   00000167 R      03
COND1_H                   00000128 RG     03
COND1_T                   0000011C R      03
COND1_TAB                 00000129 R      03
COND2                     0000028D RG     04
COND2X                    000002AF R      04
COND2_C                 = 00000000
COND2_CLEANUP             000002B0 RG     04
COND2_E                   000001A9 R      03
COND2_H                   00000188 RG     03
COND2_T                   00000173 R      03
COND2_TAB                 00000189 R      03
COND3                     000002B1 RG     04
COND3X                    000002D8 R      04
COND3_C                 = 00000000
COND3_CLEANUP             000002D9 RG     04
COND3_E                   00000245 R      03
COND3_H                   000001C6 RG     03
COND3_T                   000001B1 R      03
COND3_TAB                 000001C7 R      03
COND4                     000002DA RG     04
COND4_C                 = 00000014
COND4_CLEANUP             000002DB RG     04
COND4_H                   00000255 RG     03
COND4_T                   00000255 R      03
COND4_TAB                 00000255 R      03
COND5                     000002DC RG     04
COND5_C                 = 00000014
COND5_CLEANUP             000002DD RG     04
COND5_H                   00000256 RG     03
COND5_T                   00000256 R      03
COND5_TAB                 00000256 R      03
CONFLICT                  ********      X 04
CREATED_PID               00000118 R      03
CREATED_PRN               00000059 R      02
CREATING_PID              00000114 R      03
CTL$GL_PRD                ********    X   04
DESC                    = 00000010  G
DEST_PIDADR               0000010C R      03
DIB$R_LENGTH            = 00000074
DIB$W_UNIT              = 0000000C
EFLAG                     ********    X   04
EQUIV_PID                 0000009E R      02
EXPV                      ********    X   04
FAO_DESC                  ********    X   04
FAO_LEN                   ********    X   04
FORM_CONDS                000002DE RG     04
FORM_CONDSX               000003C7 R      04
IMAGNAM                   0000006D R      02
IO$_READVBLK              ********    X   04
LOG$C_SYSTEM            = 00000000
LOGNAM_PID                0000008C R      02
LONG                    = 00000004  G
MBXBUFF                   0000008C R      03
MBXCHAN                   00000008 R      03
MBXCHANINFO               0000000C R      03
MBXUNIT                   00000088 R      03
MOD_MSG_CODE              ********    X   04
MOD_MSG_PRINT             ********    X   04
MSG$_DEL[PROC           = 00000003
MSG1_INP_CTL              00000019 R      02
MSG3_ERR_CTL              00000039 RG     02
MSG_A                     ********    X   04
MSG_B                     ********    X   04
MSG_CTXT                  ********    X   04
NOTARG                  = 00000000  G
NULL                    = 00000014  G
ONES                      ********    X   04
ONE_MIN                   00000051 R      02
OUTPUT_MSG                ********    X   04
PCB$L_OIC               = 000000BC
PCV                       ********    X   04
PHD$Q_PRIVMSK           = 00000000
PRIVMASK                  00000000 R      03
PRIV_ARGS               = 00000002
PROCESS_ERR               ********    X   04
QUAD                    = 00000008  G
RECV                      ********    X   04
REST_REGS                 ********    X   04
SAVE_REGS                 ********    X   04
SCH$GL_CURPCB             ********    X   04
SS$_NORMAL                ********    X   04
SUCCESS                   ********    X   04
SYS$CANWAK                ********   GX   04
SYS$CMKRNL                ********   GX   04
SYS$CRELOG                ********   GX   04
SYS$CREMBX                ********   GX   04
SYS$CREPRC                ********   GX   04
SYS$DELLOG                ********   GX   04
SYS$DELMBX                ********   GX   04
SYS$DELPRC                ********   GX   04
SYS$FAO                   ********    X   04
SYS$GETCHN                ********   GX   04
SYS$HIBER                 ********   GX   04
```

```
SYS$QIOW                        ********  GX    04
SYS$SCHDWK                      ********  GX    04
SYS$SETPRN                      ********  GX    04
SYS$SETPRV                      ********  GX    04
SYS$WAKE                        ********  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                      00000268 RG     04
TM_SETUP                        00000000 RG     04
VERIFY                          000003C8 RG     04
VERIFYX                         000007C3 R      04
VFY_CLEANUP                     000007C4 RG     04
WORD                          = 00000002  G
WRITE_MSG2                      ********   X    04
ZEROPID                         00000110 R      03
```

+----------------+
! 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	000000A6 (166.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	00000257 (599.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS36	000007D4 (2004.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

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

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.10	00:00:00.65
Command processing	109	00:00:00.61	00:00:03.44
Pass 1	499	00:00:10.67	00:00:24.33
Symbol table sort	0	00:00:00.88	00:00:01.25
Pass 2	214	00:00:02.57	00:00:04.16
Symbol table output	17	00:00:00.12	00:00:00.12
Psect synopsis output	5	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	877	00:00:14.99	00:00:34.00

The working set limit was 900 pages.
56454 bytes (111 pages) of virtual memory were used to buffer the intermediate code.
There were 40 pages of symbol table space allocated to hold 577 non-local and 52 local symbols.
633 source lines were read in Pass 1, producing 26 object records in Pass 2.
51 pages of virtual memory were used to define 41 macros.

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

Macro library name                       Macros defined
------------------                       --------------
_$255$DUA28:[SHRLIB]UETP.MLB;1                         8
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                         3
_$255$DUA28:[SYSLIB]STARLET.MLB;2                     27
TOTALS (all libraries)                                38

1000 GETS were required to define 38 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:SATSSS36/OBJ=OBJ$:SATSSS36 MSRC$:SATSSS36/UPDATE=(ENH$:SATSSS36)+EXECML$/LIB+SHRLIB$:UETP/LIB
```

SATSSS35
LIS

SATSSS26
LIS

SATSSS38
LIS

SATSS530
LIS

SATSSS37
LIS

SATSSS39
LIS

SATSSS36
LIS