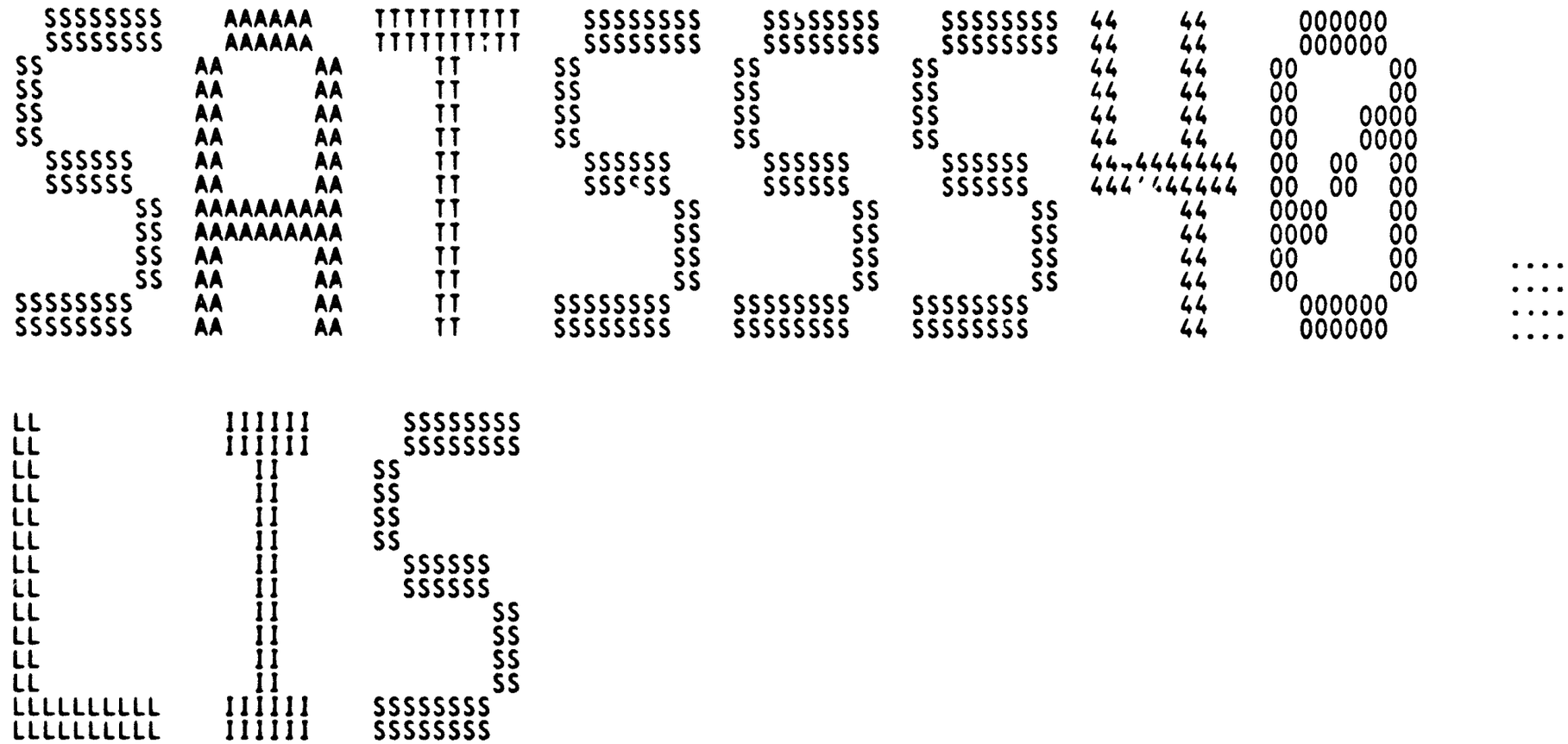
```
UUU          UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSS  YYY          YYY
UUU          UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSS  YYY          YYY
UUU          UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP        SSSSSSSSSSS  YYY          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 SSSSSSS SSSSSSSS 44 44 000000
SSSSSSSS AAAAAA TTTTTTTTTT SSSSSSSS SSSSSSSS SSSSSSSS 44 44 000000
SS AA AA TT SS SS SS 44 44 00 00
SS AA AA TT SS SS SS 44 44 00 00
SS AA AA TT SS SS SS 44 44 00 0000
SS AA AA TT SS SS SS 44 44 00 0000
SSSSSS AA AA TT SSSSSS SSSSSS SSSSSS 44 444444 00 00 00
SSSSSS AA AA TT SSSSSS SSSSSS SSSSSS 44 444444 00 00 00
SS AAAAAAAAAA TT SS SS SS 44 0000 00
SS AAAAAAAAAA TT SS SS SS 44 0000 00
SS AA AA TT SS SS SS 44 00 00
SS AA AA TT SS SS SS 44 00 00
SSSSSSSS AA AA TT SSSSSSSS SSSSSSSS SSSSSSSS 44 000000
SSSSSSSS AA AA TT SSSSSSSS SSSSSSSS SSSSSSSS 44 000000
LL IIIIII SSSSSSSS
LL IIIIII SSSSSSSS
LL II SS
LL II SS
LL II SS
LL II SS
LL II SSSSSS
LL II SSSSSS
LL II SS
LL II SS
LL II SS
LL II SS
LLLLLLLLLL IIIIII SSSSSSSS
LLLLLLLLLL IIIIII SSSSSSSS

```
0000     1          .TITLE  SATSSS40 SATS SYSTEM SERVICE TESTS  $WAKE (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 SATSSS40 TO TEST SUCCESSFUL
0000    36 ; OPERATION OF THE $WAKE SYSTEM SERVICE. THE SERVICE IS INVOKED
0000    37 ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38 ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39 ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40 ; CHECKING FOR AN SS$_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: OCT, 1977
0000    47 ;
0000    48 ; MODIFIED BY:
0000    49 ;
0000    50 ;               VERSION 1.5 : 25-MAY-79
0000    51 ; 01  LDJ  10/11/79     Fixed bug caused by DIB$K_LENGTH change ACG052.RNO mem
0000    52 ;
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          $DIBDEF                         ; DEVICE INFO BLOCK OFFSETS
0000     64 ;
0000     65 ; MACROS:
0000     66 ;
0000     67 ;
0000     68 ; EQUATED SYMBOLS:
0000     69 ;
0000     70 ;
0000     71 ; OWN STORAGE:
0000     72 ;
```

```
                                                H  1
00000000     74          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
     0000    75 TEST_MOD_NAME::  STRING  C,<SATSSS40>     ; TEST MODULE NAME
     0009    76 TEST_MOD_NAME_D: STRING  I,<SATSSS40>     ; TEST MODULE NAME DESCRIPTOR
     0019    77 MSG1_INP_CTL:    STRING  I,<  SSWAK!4ZW: CONDITIONS:>
     0039    78                                           ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
     0039    79 MSG3_ERR_CTL::   STRING  I,< *SSWAK!4ZW: !AS>
     0051    80                                           ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
     0051    81 SUBJPRN:         STRING  I,<SATSSS40_CRE> ; PROCESS & MBX NAME FOR CREATED PROCESS
     0065    82 IMAGNAM:         STRING  I,<SYSTST$RES:SATSUT06.EXE> ; IMAGE NAME FOR CREATED PROC
     0084    83 QUOTALIST:       $QUOTA  CPULM,0          ; INFINITE CPU
     0089    84                  $QUOTA  BYTLM,512        ; BYTE LIMIT FOR BUFFERED I/O
     008E    85                  $QUOTA  FILLM,2          ; OPEN FILE COUNT LIMIT
     0093    86                  $QUOTA  PGFLQUOTA,10     ; PAGING FILE QUOTA
     0098    87                  $QUOTA  PRCLM,2          ; SUBPROCESS QUOTA
     009D    88                  $QUOTA  TQELM,3          ; TIMER QUEUE ENTRY QUOTA
     00A2    89                  $QUOTA  LISTEND          ; DEFINES END OF LIST
```

```
                                                I  1
        00000000     91          .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008  0000     92 PRIVMASK:        .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000C  0008     93 MBXCHAN:         .BLKL   1               ; CHAN. NO. FOR MAILBOX FOR CREATED PROCESS
          000C     94 MBXCHANINFO:                             ; CHANNEL INFO RETURNED BY GETCHN
00000074  000C     95                  .LONG   DIB$K_LENGTH
00000014' 0010     96                  .ADDRESS .+4
00000088  0014     97                  .BLKB   DIB$K_LENGTH
0000008C  0088     98 MBXUNIT:         .BLKL   1               ; SAVE AREA FOR MAILBOX UNIT NUMBER
          008C     99 MBXBUFF:         STRING  0,120           ; MAILBOX BUFFER FOR CREATED PROCESS
00000110  010C    100 DEST_PIDADR:     .BLKL   1               ; DESTINATION PID ADDR, WRITTEN BY S.S.
00000114  0110    101 ZEROPID:         .BLKL   1               ; PID OF ZEROES
00000000  0114    102 SELFPID:         .LONG   0               ; PID OF THIS PROCESS
0000011C  0118    103 CREPID:          .BLKL   1               ; PID OF CREATED PROCESS
00000120  011C    104 SUBJPID:         .BLKL   1               ; PID OF SUBJECT PROCESS (SELF OR OTHER)
```

```
                                J 1
                     0120   106          .SBTTL  CONDITION TABLES
                     0120   107 ;
                     0120   108 ;        ***** CONDITION TABLES FOR WAKE SYSTEM SERVICE *****
                     0120   109 ;
                     0120   110          COND    1,NOTARG,<PID ADDRESS>,-
                     0120   111            <NOT SPECIFIED>,-
                     0120   112            <SPECIFIED, NON-ZERO>,-
                     0120   113            <SPECIFIED, ZERO>,-
                     0120   114
00000000'            016B   115              .ADDRESS    0
0000011C'            016F   116              .ADDRESS    SUBJPID
00000110'            0173   117              .ADDRESS    ZEROPID
                     0177   118 ;
                     0177   119          COND    2,NOTARG,<PROCESS NAME ADDRESS>,-
                     0177   120            <SPECIFIED>,-
                     0177   121            <NOT SPECIFIED>,-
                     0177   122
00000051'            01AD   123              .ADDRESS    SUBJPRN
00000000'            01B1   124              .ADDRESS    0
                     01B5   125 ;
                     01B5   126          COND    3,NOTARG,<PROCESS TYPE>,-
                     01B5   127            <SELF>,-
                     01B5   128            <SUBPROCESS>,-
                     01B5   129            <DETACHED, DIFFERENT GROUP>,-
                     01B5   130            <DETACHED, SAME GROUP, SAME MEMBER>,-
                     01B5   131            <DETACHED, SAME GROUP, DIFFERENT MEMBER>,-
                     01B5   132
FFFFFFFF             024A   133              .LONG       ^XFFFFFFFF ; PSEUDO-UIC
00000000             024E   134              .LONG       0          ; PSEUDO-UIC
00000256             0252   135              .BLKL       1          ; UIC
0000025A             0256   136              .BLKL       1          ; UIC
0000025E             025A   137              .BLKL       1          ; UIC
                     025E   138 ;
                     025E   139          COND    4,NULL
                     025F   140
                     025F   141          COND    5,NULL
                     0260   142
                00000000    143          .PSECT  SATSSS40,RD,WRT,EXE
```

```
                        0000   145          .SBTTL  TM_SETUP, TM_CLEANUP
                        0000   146 ;++
                        0000   147 ; FUNCTIONAL DESCRIPTION:
                        0000   148 ;
                        0000   149 ;               TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                        0000   150 ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                        0000   151 ; TEST MODULE EXECUTION.
                        0000   152 ;
                        0000   153 ; CALLING SEQUENCE:
                        0000   154 ;
                        0000   155 ;       BSBW TM_SETUP   BSBW TM_CLEANUP
                        0000   156 ;
                        0000   157 ; INPUT PARAMETERS:
                        0000   158 ;
                        0000   159 ;       NONE
                        0000   160 ;
                        0000   161 ; IMPLICIT INPUTS:
                        0000   162 ;
                        0000   163 ;       NONE
                        0000   164 ;
                        0000   165 ; OUTPUT PARAMETERS:
                        0000   166 ;
                        0000   167 ;       NONE
                        0000   168 ;
                        0000   169 ; IMPLICIT OUTPUTS:
                        0000   170 ;
                        0000   171 ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                        0000   172 ;                  ALL PRIVILEGES ACQUIRED.
                        0000   173 ;
                        0000   174 ; COMPLETION CODES:
                        0000   175 ;
                        0000   176 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                        0000   177 ;
                        0000   178 ; SIDE EFFECTS:
                        0000   179 ;
                        0000   180 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                        0000   181 ;       (VIA RSB) IF ERROR ENCOUNTERED.
                        0000   182 ;
                        0000   183 ;--
                        0000   184
                        0000   185
                        0000   186
                        0000   187 TM_SETUP::
                  52    D4    0000   188          CLRL    R2                      ; INITIALIZE
                  53    D4    0002   189          CLRL    R3                      ; .. CONDITION
                  54    D4    0004   190          CLRL    R4                      ; .... TABLE
                  55    D4    0006   191          CLRL    R5                      ; ...... INDEX
                  56    D4    0008   192          CLRL    R6                      ; ........ REGISTERS
              FFF3'   30    000A   193          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE BEGIN MSG
00000000'EF   00000000'EF   DE    000D   194          MOVAL   TEST_MOD_SUCC,TMD_ADDR  ; ASSUME END MSG WILL SHOW SUCCESS
   03   00   00000000'8F   F0    0018   195          INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
             00000000'EF         0020
                               0025   196          MODE    TO,S$,KRNL              ; KERNEL MODE TO ACCESS PHD
        59   00000000'9F   D0    0048   197          MOVL    @#CTL$GL_PHD,R9         ; GET PROCESS HEADER ADDRESS
        000000C0'EF   69   DE    004F   198          MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                               0056   199          MODE    FROM,S$ ; BACK TO USER MODE
                               0057   200          PRIV    ADD,ALL                 ; GET ALL PRIVILEGES
```

```
                     0077   201          $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                     0084   202          SS_CHECK  NORMAL                ; CHECK STATUS CODE RETURNED FROM SETPRN
                     00B2   203          $WAKE_S SELFPID                 ; GET MY PID
                     00C1   204          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                     00EF   205          $HIBER_S                        ; UNDO ABOVE WAKE
                     00F6   206          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                     0124   207 ;
                     0124   208 ; THE FOLLOWING CODE ESTABLISHES UIC'S IN THE CONDITION 3 TABLE
                     0124   209 ;
                     0124   210          MODE    TO,20$,KRNL             ; KERNEL MODE TO ACCESS PCB
     59  00000000'9F  D0     0147   211          MOVL    @#SCH$GL_CURPCB,R9      ; GET CURRENT PCB ADDRESS
         59  00BC C9  D0     014E   212          MOVL    PCB$L_UIC(R9),R9        ; PICK UP UIC FROM PCB
                     0153   213          MODE    FROM,20$                ; ... AND GET BACK TO USER MODE
                     0154   214 ;
                     0154   215 ; R9 NOW CONTAINS "MY" UIC
                     0154   216 ;
            5A   02  9A     0154   217          MOVZBL  #2,R10                  ; GET COND3 TABLE INDEX NUMBER INTO A REG
     59  00010000  8F  C1     0157   218          ADDL3   #^X10000,R9,COND3_E[R10] ; PUT DIFF GROUP UIC INTO 3RD TABLE ELT
        0000024A'EF4A        015E
            5A   D6     0164   219          INCL    R10                     ; POINT TO 4TH COND3 TABLE ELEMENT
0000024A'EF4A   59   D0     0166   220          MOVL    R9,COND3_E[R10]         ; PUT MY UIC INTO TABLE
            5A   D6     016E   221          INCL    R10                     ; POINT TO 5TH COND3 TABLE ELEMENT
0000024A'EF4A   59   01   C1     0170   222          ADDL3   #1,R9,COND3_E[R10]      ; PUT DIFF MEMBER UIC INTO THE TABLE
                     0179   223          $CREMBX_S CHAN=MBXCHAN, LOGNAM=SUBJPRN, - ; GET MAILBOX FOR PROCESS
                     0179   224                    MAXMSG=#120, PROMSK=#0, BUFQUO=#240
                     019E   225          SS_CHECK NORMAL                 ; CHECK NORMAL COMPLETION
                     01CC   226          $GETCHN_S CHAN=MBXCHAN, -       ; GET CHAN INFO (UNIT NUMBER)
                     01CC   227                    PRIBUF=MBXCHANINFO
                     01E6   228          SS_CHECK NORMAL                 ; CHECK NORMAL COMPLETION
00000088'EF  00000020'EF  3C     0214   229          MOVZWL  MBXCHANINFO+8+DIB$W_UNIT,MBXUNIT ; SAVE MAILBOX UNIT NUMBER
                  05     021F   230          RSB                             ; RETURN TO MAIN ROUTINE
                     0220   231 TM_CLEANUP::
                     0220   232          $DELMBX_S  MBXCHAN              ; DELETE TERMINATION MAILBOX
                FDCF'  30     022E   233          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE END MSG
                  05     0231   234          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 SOBROUTINE, 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      0232    RSB                             ; RETURN TO MAIN ROUTINE
                COND1_CLEANUP::
05      0233    RSB                             ; RETURN TO MAIN ROUTINE
                COND2::
05      0234    RSB                             ; RETURN TO MAIN ROUTINE
                COND2_CLEANUP::
05      0235    RSB                             ; RETURN TO MAIN ROUTINE
```

```
                                0236   293 COND3::
0000016B'EF42   0000011C'8F     D1 0236   294          CMPL    #SUBJPID,COND1_E[R2]    ; NON-ZERO PID SPECIFIED ?
                          20    13 0242   295          BEQLU   10$                     ; YES -- PROCESS IS "OTHER"
                0000C1AD'EF43   D5 0244   296          TSTL    COND2_E[R3]             ; IS PROCESS NAME SPECIFIED ?
                          07    13 024B   297          BEQL    5$                      ; NO -- SUBJECT PROCESS IS "SELF"
                       02   54    D1 024D   298          CMPL    R4,#2                   ; DOES CONDITION 3 SPECIFY DIFFERENT GROUP ?
                          20    13 0250   299          BEQL    20$                     ; YES -- PROCESS NAME FOR DIFF GROUP IS CONF
                          10    11 0252   300          BRB     10$                     ; NO -- MAKE SURE COND 3 SPECIFIES "OTHER"
                                0254   301 5$:
                                0254   302 ;
                                0254   303 ; PROCESS IS "SELF"
                                0254   304 ;
0000024A'EF44   00000000'EF     D1 0254   305          CMPL    ONES,COND3_E[R4]        ; DOES CONDITION 3 SPECIFY "SELF" ?
                          1B    13 0260   306          BEQLU   COND3X                  ; YES -- THEN ALL 3 CONDIT'NS ARE CONSISTENT
                          0E    11 0262   307          BRB     20$                     ; NO -- INDICATE CONFLICT & GET OUT
                                0264   308 10$:
                                0264   309 ;
                                0264   310 ; PROCESS IS "OTHER"
                                0264   311 ;
0000024A'EF44   00000000'EF     D1 0264   312          CMPL    ONES,COND3_E[R4]        ; DOES CONDITION 3 SPECIFY "SELF" ?
                          0B    12 0270   313          BNEQU   COND3X                  ; NO -- THEN ALL 3 CONDITIONS ARE CONSISTENT
                                0272   314 20$:
  00000000'EF   00000000'EF     90 0272   315          MOVB    ONES,CONFLICT           ; YES -- INDICATE CONFLICT
                                0275   316 COND3X:
                             05 0275   317          RSB                             ; RETURN TO MAIN ROUTINE
                                0276   318 COND3_CLEANUP::
                             05 0276   319          RSB                             ; RETURN TO MAIN ROUTINE
                                0277   320 COND4::
                             05 0277   321          RSB                             ; RETURN TO MAIN ROUTINE
                                0278   322 COND4_CLEANUP::
                             05 0278   323          RSB                             ; RETURN TO MAIN ROUTINE
                                0279   324 COND5::
                             05 0279   325          RSB                             ; RETURN TO MAIN ROUTINE
                                027A   326 COND5_CLEANUP::
                             05 027A   327          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                                0283   329          .SBTTL  FORM_CONDS
                                0283   330 ;++
                                0283   331 ; FUNCTIONAL DESCRIPTION:
                                0283   332 ;
                                0283   333 ;               FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                                0283   334 ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                                0283   335 ;
                                0283   336 ; CALLING SEQUENCE:
                                0283   337 ;
                                0283   338 ;       BSBW FORM_CONDS
                                0283   339 ;
                                0283   340 ; INPUT PARAMETERS:
                                0283   341 ;
                                0283   342 ;       NONE
                                0283   343 ;
                                0283   344 ; IMPLICIT INPUTS:
                                0283   345 ;
                                0283   346 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                                0283   347 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                                0283   348 ;       FOR X = 1,2,3,4,5 :
                                0283   349 ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                                0283   350 ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                                0283   351 ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                                0283   352 ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                                0283   353 ;
                                0283   354 ; OUTPUT PARAMETERS:
                                0283   355 ;
                                0283   356 ;       NONE
                                0283   357 ;
                                0283   358 ; IMPLICIT OUTPUTS:
                                0283   359 ;
                                0283   360 ;       NONE
                                0283   361 ;
                                0283   362 ; COMPLETION CODES:
                                0283   363 ;
                                0283   364 ;       NONE
                                0283   365 ;
                                0283   366 ; SIDE EFFECTS:
                                0283   367 ;
                                0283   368 ;       NONE
                                0283   369 ;
                                0283   370 ;--
                                0283   371
                                0283   372
                                0283   373
                                0283   374 FORM_CONDS::
                                0283   375          $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                                02A2   376                                          ; FORMAT CONDITIONS HEADER MSG
                         FD5B'  30     02A2   377          BSBW    OUTPUT_MSG       ; ... AND PRINT IT
                      14   00  91     02A5   378          CMPB    #COND1_C,#NULL   ; IS CONDITION 1 NULL ?
                           03  12     02A8   379          BNEQU   10$              ; NO -- CONTINUE
                         00BF  31     02AA   380          BRW     FORM_CONDSX      ; YES -- SUBROUTINE IS FINISHED
                                      02AD   381 10$:
  00000000'EF    00000120'EF  DE     02AD   382          MOVAL   COND1_T,MSG_A    ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF    0000012D'EF42  D0     02B8   383          MOVL    COND1_TAB[R2],MSG_B ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
          00000000'EF   00    90     02C4   384          MOVB    #COND1_C,MSG_CTXT ; SAVE CONDITION 1 CONTEXT FOR FAO
                                      02CB   385          MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                     FD32'   30  02CB   386          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 1 MSG
                  14   00    91  02CE   387          CMPB    #COND2_C,#NULL                  ; IS CONDITION 2 NULL ?
                       03    12  02D1   388          BNEQU   20$                             ; NO -- CONTINUE
                     0096    31  02D3   389          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                             02D6   390 20$:
  00000000'EF   00000177'EF    DE  02D6   391          MOVAL   COND2_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF   0000018D'EF43    D0  02E1   392          MOVL    COND2_TAB[R3],MSG_B             ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
        00000000'EF   00    90  02ED   393          MOVB    #COND2_C,MSG_CTXT               ; SAVE CONDITION 2 CONTEXT FOR FAO
                             02F4   394          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                     FD09'   30  02F4   395          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 2 MSG
                  14   00    91  02F7   396          CMPB    #COND3_C,#NULL                  ; IS CONDITION 3 NULL ?
                       03    12  02FA   397          BNEQU   30$                             ; NO -- CONTINUE
                     006D    31  02FC   398          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                             02FF   399 30$:
  00000000'EF   000001B5'EF    DE  02FF   400          MOVAL   COND3_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF   000001C3'EF44    D0  030A   401          MOVL    COND3_TAB[R4],MSG_B             ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
        00000000'EF   00    90  0316   402          MOVB    #COND3_C,MSG_CTXT               ; SAVE CONDITION 3 CONTEXT FOR FAO
                             031D   403          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                     FCE0'   30  031D   404          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 3 MSG
                  14   14    91  0320   405          CMPB    #COND4_C,#NULL                  ; IS CONDITION 4 NULL ?
                       47    13  0323   406          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   0000025E'EF    DE  0325   407          MOVAL   COND4_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF   0000025E'EF45    D0  0330   408          MOVL    COND4_TAB[R5],MSG_B             ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
        00000000'EF   14    90  033C   409          MOVB    #COND4_C,MSG_CTXT               ; SAVE CONDITION 4 CONTEXT FOR FAO
                             0343   410          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                     FCBA'   30  0343   411          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 4 MSG
                  14   14    91  0346   412          CMPB    #COND5_C,#NULL                  ; IS CONDITION 5 NULL ?
                       21    13  0349   413          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   0000025F'EF    DE  034B   414          MOVAL   COND5_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF   0000025F'EF46    D0  0356   415          MOVL    COND5_TAB[R6],MSG_B             ; SAVE ADDR OF COND 5 CUPR TEXT ELT FOR FAO
        00000000'FF   14    90  0362   416          MOVB    #COND5_C,MSG_CTXT               ; SAVE CONDITION 5 CONTEXT FOR FAO
                             0369   417          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                     FC94'   30  0369   418          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 5 MSG
                             036C   419 FORM_CONDSX:
                       05    03C   420          RSB                                     ; RETURN TO CALLER
```

```
                        D 2
036D    422          .SBTTL  VERIFY
036D    423 ;++
036D    424 ; FUNCTIONAL DESCRIPTION:
036D    425 ;
036D    426 ;                   VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
036D    427 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
036D    428 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
036D    429 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
036D    430 ; ($WAKE). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
036D    431 ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
036D    432 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
036D    433 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
036D    434 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
036D    435 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO
036D    436 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
036D    437 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
036D    438 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
036D    439 ;
036D    440 ; CALLING SEQUENCE:
036D    441 ;
036D    442 ;       BSBW VERIFY
036D    443 ;
036D    444 ; INPUT PARAMETERS:
036D    445 ;
036D    446 ;       NONE
036D    447 ;
036D    448 ; IMPLICIT INPUTS:
036D    449 ;
036D    450 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
036D    451 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
036D    452 ;       FOR X = 1,2,3,4,5 :
036D    453 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
036D    454 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
036D    455 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
036D    456 ;                 FOR CONDX_E.
036D    457 ;
036D    458 ; OUTPUT PARAMETERS:
036D    459 ;
036D    460 ;       NONE
036D    461 ;
036D    462 ; IMPLICIT OUTPUTS:
036D    463 ;
036D    464 ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
036D    465 ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
036D    466 ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
036D    467 ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
036D    468 ;       ERRORS.
036D    469 ;
036D    470 ; COMPLETION CODES:
036D    471 ;
036D    472 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
036D    473 ;
036D    474 ; SIDE EFFECTS:
036D    475 ;
036D    476 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
036D    477 ;       (VIA RSB) IF ERROR ENCOUNTERED.
036D    478 ;
```

```
                                                E 2
                                036D   479 ;--
                                036D   480
                                036D   481
                                036D   482
                                036D   483 VERIFY::
                00000000'EF  95 036D   484          TSTB    CFLAG                   ; SHOULD CONDITIONS BE PRINTED ?
                          03 13 0373   485          BEQL    5$                      ; NO -- CONTINUE
                        FF0B 30 0375   486          BSBW    FORM_CONDS              ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                                0378   487 5$:
  0000011C'EF   00000114'EF  D0 0378   488          MOVL    SELFPID,SUBJPID         ; ASSUME THE SUBJECT PID IS SELF
                00000110'EF  D4 0383   489          CLRL    ZEROPID                 ; CLEAR ZERO PID
0000024A'EF44   00000000'EF  D1 0389   490          CMPL    ONES,COND3_E[R4]        ; IS PROCESS FOR THIS TEST CASE SELF ?
                          03 12 0395   491          BNEQU   7$                      ; NO -- CONTINUE
                        0074 31 0397   492          BRW     10$                     ; YES -- DON'T CREATE A PROCESS
                                039A   493 7$:
                                039A   494          $CREPRC_S PIDADR=CREPID, PRCNAM=SUBJPRN, -
                                039A   495                    UIC=COND3_E[R4], IMAGE=IMAGNAM, -
                                039A   496                    QUOTA=QUOTALIST,MBXUNT=MBXUNIT
                                03D5   497                                            ; CREATE THE SUBJECT PROCESS
                                03D5   498          SS_CHECK NORMAL                   ; ... AND MAKE SURE IT CREATED OK
  0000011C'EF   00000118'EF  D0 0403   499          MOVL    CREPID,SUBJPID          ; MAKE THE SUBJCT PID = THE ONE JUST CREATED
                                040E   500 10$:
0000010C'EF   0000016B'EF42  D0 040E   501          MOVL    COND1_E[R2],DEST_PIDADR ; GET PID ADDRESS OUT OF TABLE
          59  000001AD'EF43  D0 041A   502          MOVL    COND2_E[R3],R9          ; PRCNAM ADDR INTO REG FOR INDIRECT REF'RNCE
                                0422   503 ;
                                0422   504 ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                                0422   505 ;
                                0422   506          $WAKE_S PIDADR=@DEST_PIDADR, PRCNAM=(R9)
          00000000'8F    50 D1 0431   507          CMPL    R0,#SS$_NORMAL          ; CODE RECEIVED = CODE EXPECTED ?
                          5F 13 0438   508          BEQLU   18$                     ; YES -- CONTINUE
  00000000'EF   00000000'8F  D0 043A   509          MOVL    #SS$_NORMAL,EXPV        ; NO -- LOAD UP EXPECTED AND ...
          00000000'EF    50 D0 0445   510          MOVL    R0,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                                044C   511          ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM WAKE>
                                0499   512 18$:
                0000010C'EF  D5 0499   513          TSTL    DEST_PIDADR             ; PID RETURNED BY WAKE ?
                          66 13 049F   514          BEQL    20$                     ; NO -- KEEP GOING
0000010C'FF     0000011C'EF  D1 04A1   515          CMPL    SUBJPID,@DEST_PIDADR    ; YES -- IS IT THE CORRECT ONE ?
                          59 13 04AC   516          BEQL    20$                     ; YES -- CONTINUE
00000000'EF     0000011C'EF  D0 04AE   517          MOVL    SUBJPID,EXPV            ; NO --LOAD UP EXPECTED AND
00000000'EF     0000010C'FF  D0 04B9   518          MOVL    @DEST_PIDADR,RECV       ; ... RECEIVED VALUES, THEN EXIT
                                04C4   519          ERR_EXIT LONG,<INCORRECT PID RETURNED BY WAKE>
                                0507   520 20$:
0000011C'EF     0000118'EF   D1 0507   521          CMPL    CREPID,SUBJPID          ; WAS A PROCESS CREATED ?
                          37 13 0512   522          BEQLU   30$                     ; YES -- GO WAIT FOR IT TO COMPLETE
                                0514   523          $HIBER_S                        ; NO -- OFFSET SUBJECT WAKE WITH HIBER
                                051B   524          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                          57 11 0549   525          BRB     VERIFYX                 ; ... AND GO EXIT
                                054B   526 30$:
                                054B   527          $QIOW_S CHAN=MBXCHAN, FUNC=#IO$_READVBLK, -
                                054B   528                  P1=MBXBUFF+8, P2=MBXBUFF
                                0574   529                                          ; WAIT FOR CREATED PROCESS TO SEND MAIL
                                0574   530          SS_CHECK NORMAL                 ; CHECK FOR NORMAL STATUS CODE
                                05A2   531 VERIFYX:
                             05 05A2   532          RSB                             ; RETURN TO CALLER
```

```
                        05A3    534          .SBTTL  VFY_CLEANUP
                        05A3    535 ;++
                        05A3    536 ; FUNCTIONAL DESCRIPTION:
                        05A3    537 ;
                        05A3    538 ;               VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
                        05A3    539 ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
                        05A3    540 ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
                        05A3    541 ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
                        05A3    542 ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
                        05A3    543 ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
                        05A3    544 ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
                        05A3    545 ; POSSIBLY DISCOVERING A SECOND ERROR.
                        05A3    546 ;
                        05A3    547 ; CALLING SEQUENCE:
                        05A3    548 ;
                        05A3    549 ;       BSBW VFY_CLEANUP
                        05A3    550 ;
                        05A3    551 ; INPUT PARAMETERS:
                        05A3    552 ;
                        05A3    553 ;       NONE
                        05A3    554 ;
                        05A3    555 ; IMPLICIT INPUTS:
                        05A3    556 ;
                        05A3    557 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                        05A3    558 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                        05A3    559 ;       FOR X = 1,2,3,4,5 :
                        05A3    560 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
                        05A3    561 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
                        05A3    562 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
                        05A3    563 ;                 FOR CONDX_E.
                        05A3    564 ;
                        05A3    565 ; OUTPUT PARAMETERS:
                        05A3    566 ;
                        05A3    567 ;       NONE
                        05A3    568 ;
                        05A3    569 ; IMPLICIT OUTPUTS:
                        05A3    570 ;
                        05A3    571 ;       NONE
                        05A3    572 ;
                        05A3    573 ; COMPLETION CODES:
                        05A3    574 ;
                        05A3    575 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                        05A3    576 ;
                        05A3    577 ; SIDE EFFECTS:
                        05A3    578 ;
                        05A3    579 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                        05A3    580 ;       (VIA RSB) IF ERROR ENCOUNTERED.
                        05A3    581 ;
                        05A3    582 ;--
                        05A3    583
                        05A3    584
                        05A3    585
                        05A3    586 VFY_CLEANUP::
0000011C'EF   00000118'EF D1    05A3    587          CMPL    CREPID,SUBJPID          ; WAS A PROCESS CREATED FOR THIS TEST CASE ?
                      0F  12    05AE    588          BNEQU   VFY_CLEANUPX            ; NO -- JUST EXIT
                                05B0    589          $DELPRC_S SOBJPID               ; YES -- DELETE IT
                                05BF    590 VFY_CLEANUPX:
```

```
 05  05BF    591          RSB                             ; RETURN TO CALLER
     05C0    592          .END
```

```
$$$$                    = 000004CE R      04
$$$CHARS                = 0000001E
$$$CHARS1               = 00000004
$$$CHARS2               = 0000000A
$$$CHARS3               = 00000019
$$$CHARS4               = 00000021
$$$CHARS5               = 00000026
$$$COND_A               = 00000004
$$$$STRINGS             = 00000001
$$$$STRINGS2            = 00000005
$$T1                    = 00000001
$$T2                    = 00000004
BYTE                    = 00000001   G
CFLAG                     ********     X    04
CHMRTN                    ********     X    04
CHM_CONT                  ********     X    04
COMP_SC                   ********     X    04
CONDT                     00000232 RG       04
COND1_C                 = 00000000
COND1_CLEANUP             00000233 RG       04
COND1_E                   0000016B R        03
COND1_H                   0000012C RG       03
COND1_T                   00000120 R        03
COND1_TAB                 0000012D R        03
COND2                     00000234 RG       04
COND2_C                 = 00000000
COND2_CLEANUP             00000235 RG       04
COND2_E                   000001AD R        03
COND2_H                   0000018C RG       03
COND2_T                   00000177 R        03
COND2_TAB                 0000018D R        03
COND3                     00000236 RG       04
COND3X                    0000027D R        04
COND3_C                 = 00000000
COND3_CLEANUP             0000027E RG       04
COND3_E                   0000024A R        03
COND3_H                   000001C2 RG       03
COND3_T                   000001B5 R        03
COND3_TAB                 000001C3 R        03
COND4                     0000027F RG       04
COND4_C                 = 00000014
COND4_CLEANUP             00000280 RG       04
COND4_H                   0000025E RG       03
COND4_T                   0000025E R        03
COND4_TAB                 0000025E R        03
COND5                     00000281 RG       04
COND5_C                 = 00000014
COND5_CLEANUP             00000282 RG       04
COND5_H                   0000025F RG       03
COND5_T                   0000025F R        03
COND5_TAB                 0000025F R        03
CONFLICT                  ********       X  04
CREPID                    00000118 R        03
CTL$GL_PHD                ********       X  04
DESC                    = 00000010   G
DEST_PIDADR               0000010C R        03
DIB$R_LENGTH            = 00000074

DIB$W_UNIT              = 0000000C
EFLAG                     ********     X    04
EXPV                      ********     X    04
FAO_DESC                  ********     X    04
FAO_LEN                   ********     X    04
FORM_CONDS                00000283 RG       04
FORM_CONDSX               0000036C R        04
IMAGNAM                   00000065 R        02
IOS_READVBLK              ********       X  04
LONG                    = 00000004   G
MBXBUFF                   0000008C R        03
MBXCHAN                   00000008 R        03
MBXCHANINFO               0000000C R        03
MBXUNIT                   00000C88 R        03
MOD_MSG_CODE              ********       X  04
MOD_MSG_PRINT             ********       X  04
MSGT_INP_CTL              00000019 R        02
MSG3_ERR_CTL              00000039 RG       02
MSG_A                     ********       X  04
MSG_B                     ********       X  04
MSG_CTXT                  ********       X  04
NOTARG                  = 00000000   G
NULL                    = 00000014   G
ONES                      ********     X    04
OUTPUT_MSG                ********     X    04
PCB$L_DIC               = 000000BC
PCV                       ********     X    04
PHD$Q_PRIVMSK           = 00000000
PQL$_BYTLM              = 00000003
PQL$_CPULM              = 00000004
PQL$_FILLM              = 00000006
PQL$_LISTEND            = 00000000
PQL$_PGFLQUOTA          = 00000007
PQL$_PRCLM              = 00000008
PQL$_TQELM              = 00000009
PRIVMASK                  00000000 R        03
PRIV_ARGS               = 00000002
PROCESS_ERR               ********       X  04
QUAD                    = 00000008   G
QUOTALIST                 00000084 R        02
RECV                      ********       X  04
REST_REGS                 ********       X  04
SAVE_REGS                 ********       X  04
SCH$GL_CURPCB             ********       X  04
SELFPID                   00000114 R        03
SS$_NORMAL                ********       X  04
SUBJPID                   0000011C R        03
SUBJPRN                   00000051 R        02
SUCCESS                   ********       X  04
SYS$CMKRNL                ********     GX   04
SYS$CREMBX                ********     GX   04
SYS$CREPRC                ********     GX   04
SYS$DELMBX                ********     GX   04
SYS$DELPRC                ********     GX   04
SYS$FAO                   ********       X  04
SYS$GETCHN                ********     GX   04
SYS$HIBER                 ********     GX   04
```

```
SYS$QIOW                        ********  GX    04
SYS$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                      00000220 RG     04
TM_SETUP                        00000000 RG     04
VERIFY                          0000036D RG     04
VERIFYX                         000005A2 R      04
VFY_CLEANUP                     000005A3 RG     04
VFY_CLEANUPX                    0000C5BF R      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	000000A7 (167.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	00000260 (608.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSS40	000005C0 (1472.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.06	00:00:00.33
Command processing	107	00:00:00.68	00:00:02.45
Pass 1	295	00:00:08.90	00:00:15.63
Symbol table sort	0	00:00:00.75	00:00:00.83
Pass 2	128	00:00:02.15	00:00:02.76
Symbol table output	17	00:00:00.12	00:00:00.14
Psect synopsis output	2	00:00:00.03	00:00:00.04
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	580	00:00:12.70	00:00:22.18

```
The working set limit was 1350 pages.
46712 bytes (92 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 487 non-local and 44 local symbols.
592 source lines were read in Pass 1, producing 24 object records in Pass 2.
46 pages of virtual memory were used to define 36 macros.
```

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

Macro library name                   Macros defined
------------------                   --------------
_$255$DUA28:[SHRLIB]UETP.MLB;1                 9
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                 2
_$255$DUA28:[SYSLIB]STARLET.MLB;2             22
TOTALS (all libraries)                        33

884 GETS were required to define 33 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS:SATSSS40/OBJ=OBJ$:SATSSS40 MSRC$:SATSSS40/UPDATE=(ENH$:SATSSS40)+EXECML$/LIB+SHRLIB$:UETP/LIB
```

SATSS552
LIS

SATSS542
LIS

SATSS540
LIS

SATSS550
LIS

SATSS544
LIS

SATSS546
LIS

SATSS543
LIS

SATSS545
LIS

SATSS541
LIS

SATSS547
LIS