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

```
  SSSSSSSS    AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS   666666      11
  SSSSSSSS    AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS   666666      11
SS           AA    AA       TT      SS         SS         SS         66          1111
SS           AA    AA       TT      SS         SS         SS         66          1111
SS           AA    AA       TT      SS         SS         SS         66            11
SS           AA    AA       TT      SS         SS         SS         66            11
  SSSSSS     AA    AA       TT        SSSSSS     SSSSSS     SSSSSS   66666666      11
  SSSSSS     AA    AA       TT        SSSSSS     SSSSSS     SSSSSS   66666666      11
        SS   AAAAAAAAAA       TT              SS         SS         SS 66    66     11
        SS   AAAAAAAAAA       TT              SS         SS         SS 66    66     11
        SS   AA    AA         TT              SS         SS         SS 66    66     11          ....
        SS   AA    AA         TT              SS         SS         SS 66    66     11          ....
SSSSSSSS     AA    AA         TT      SSSSSSSS   SSSSSSSS   SSSSSSSS     666666  111111        ....
SSSSSSSS     AA    AA         TT      SSSSSSSS   SSSSSSSS   SSSSSSSS     666666  111111        ....

LL             IIIIII      SSSSSSSS
LL             IIIIII      SSSSSSSS
LL               II      SS
LL               II      SS
LL               II      SS
LL               II      SS
LL               II        SSSSSS
LL               II        SSSSSS
LL               II              SS
LL               II              SS
LL               II              SS
LL               II              SS
LLLLLLLLLL     IIIIII    SSSSSSSS
LLLLLLLLLL     IIIIII    SSSSSSSS
```

Table of contents

```
0000     1          .TITLE  SATSSS61 SATS SYST SERV TESTS  $SCH/CANWAK (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 SATSSS61 TO TEST SUCCESSFUL
0000    36 ; OPERATION OF THE $SCH/CANWAK SYSTEM SERVICE. THE SERVICE IS INVOKED
0000    37 ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38 ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39 ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40 ; CHECKING FOR AN SS$_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: APR, 1977
0000    47 ;
0000    48 ; MODIFIED BY:
0000    49 ;
0000    50 ;       V03-001 LDJ0001         Larry D. Jones,         23-Jun-1983
0000    51 ;                Removed the quota list to force the use of the
0000    52 ;                default sysboot quotas.
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 ;
            00989680 0000     70 ONE_SEC          = 10*1000*1000              ; 10 MILLION 100-NANOSECOND UNITS (OR 1 SEC)
                     0000     71 ;
                     0000     72 ; OWN STORAGE:
                     0000     73 ;
```

```
                                                E 10
00000000     75          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
0000     76 TEST_MOD_NAME:: STRING  C,<SATSSS61>     ; TEST MODULE NAME
0009     77 TEST_MOD_NAME_D: STRING I,<SATSSS61>     ; TEST MODULE NAME DESCRIPTOR
0019     78 MSG1_INP_CTL:   STRING  I,<  SSSCW!4ZW: CONDITIONS:>
0039     79                                          ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
0039     80 MSG3_ERR_CTL::  STRING  I,< *SSSCW!4ZW: !AS>
0051     81                                          ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
0051     82 SUBJPRN:        STRING  I,<SATSSS61_CRE> ; PROCESS & MBX NAME FOR CREATED PROCESS
0065     83 IMAGNAM:        STRING  I,<SYSTST$RES:SATSUT07.EXE> ; IMAGE NAME FOR CREATED PROC
31 36 53 53 0000008C'010E0000' 0084  84 CLUSTER:        .ASCID  /SS61/              ; STRING DESCRIPTOR FOR CLUSTER
0090     85                                          ; ... FOR CREATED PROCESS COMMUNICATION
0090     86 ;QUOTALIST:     $QUOTA  CPULM,0          ; INFINITE CPU
0090     87 ;               $QUOTA  BYTLM,512        ; BYTE LIMIT FOR BUFFERED I/O
0090     88 ;               $QUOTA  FILLM,2          ; OPEN FILE COUNT LIMIT
0090     89 ;               $QUOTA  PGFLQUOTA,10     ; PAGING FILE QUOTA
0090     90 ;               $QUOTA  PRCLM,2          ; SUBPROCESS QUOTA
0090     91 ;               $QUOTA  TQELM,3          ; TIMER QUEUE ENTRY QUOTA
0090     92 ;               $QUOTA  LISTEND          ; DEFINES END OF LIST
FFFFFFFF FF676980 0090     93 DELTA_1SEC:   .LONG   -ONE_SEC,-1      ; DELTA TIME VALUE FOR 1 SECOND
FFFFFFFF FECED300 0098     94 DELTA_2SEC:   .LONG   -2*ONE_SEC,-1    ; DELTA TIME VALUE FOR 2 SECONDS
FFFFFFFF FE363C80 00A0     95 DELTA_3SEC:   .LONG   -3*ONE_SEC,-1    ; DELTA TIME VALUE FOR 3 SECONDS
FFFFFFFF FA0A1F00 00A8     96 DELTA_10SEC:  .LONG   -10*ONE_SEC,-1   ; DELTA TIME VALUE FOR 10 SECONDS
FFFFFFFF FFD9DA60 00B0     97 DELTA_QSEC:   .LONG   -<ONE_SEC/4>,-1  ; DELTA TIME VALUE FOR A QUARTER-SECOND
00000000 01C9C380 00B8     98 POS_3SEC:     .LONG   3*ONE_SEC,0      ; 3 SECONDS (POSITIVE VALUE)
00C0     99 TIME_PAST:     STRING  I,<25-DEC-1973  21:46:00.00> ; A TIME IN THE PAST
```

```
        00000000     101                .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008  0000       102 PRIVMASK:      .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000C  0008       103 MBXCHAN:       .BLKL   1               ; CHAN. NO. FOR MAILBOX FOR CREATED PROCESS
          000C       104 MBXCHANINFO:                           ; CHANNEL INFO RETURNED BY GETCHN
00000074  000C       105                .LONG   DIB$K_LENGTH
00000014' 0010       106                .ADDRESS .+4
00000088  0014       107                .BLKB   DIB$K_LENGTH
0000008C  0088       108 MBXUNIT:       .BLKL   1               ; SAVE AREA FOR MAILBOX UNIT NUMBER
          008C       109 MBXBUFF:       STRING  0,120           ; MAILBOX BUFFER FOR CREATED PROCESS
00000110  010C       110 DEST_PIDADR:   .BLKL   1               ; DESTINATION PID ADDR, WRITTEN BY S.S.
00000114  0110       111 ZEROPID:       .BLKL   1               ; PID OF ZEROES
00000000  0114       112 SELFPID:       .LONG   0               ; PID OF THIS PROCESS
0000011C  0118       113 CREPID:        .BLKL   1               ; PID OF CREATED PROCESS
00000120  011C       114 SUBJPID:       .BLKL   1               ; PID OF SUBJECT PROCESS (SELF OR OTHER)
00000128  0120       115 ABS_3SEC:      .BLKQ   1               ; WILL HOLD ABS TIME VALUE FOR NOW + 3 SECS
00000130  0128       116 ABS_PAST:      .BLKQ   1               ; WILL HOLD ABS TIME VALUE FOR TIME IN PAST
00000131  0130       117 LONG_WAIT:     .BLKB   1               ; LONG WAIT INDICATOR; 0=NO LONG WAIT
```

```
                     0131   119          .SBTTL  CONDITION TABLES
                     0131   120 ;
                     0131   121 ;       ***** CONDITION TABLES FOR SCH/CANWAK SYSTEM SERVICE *****
                     0131   122 ;
                     0131   123          COND    1,NOTARG,<PID ADDRESS>,-
                     0131   124                    <NOT SPECIFIED>,-
                     0131   125                    <SPECIFIED, NON-ZERO>,-
                     0131   126                    <SPECIFIED, ZERO>,-
                     0131   127
00000000'            017C   128                      .ADDRESS    0
0000011C'            0180   129                      .ADDRESS    SUBJPID
00000110'            0184   130                      .ADDRESS    ZEROPID
                     0188   131 ;
                     0188   132          COND    2,NOTARG,<PROCESS NAME ADDRESS>,-
                     0188   133                    <SPECIFIED>,-
                     0188   134                    <NOT SPECIFIED>,-
                     0188   135
00000051'            01BE   136                      .ADDRESS    SUBJPRN
00000000'            01C2   137                      .ADDRESS    0
                     01C6   138 ;
                     01C6   139          COND    3,NOTARG,<PROCESS TYPE>,-
                     01C6   140                    <SELF>,-
                     01C6   141                    <SUBPROCESS>,-
                     01C6   142                    <DETACHED, DIFFERENT GROUP>,-
                     01C6   143                    <DETACHED, SAME GROUP, SAME MEMBER>,-
                     01C6   144                    <DETACHED, SAME GROUP, DIFFERENT MEMBER>,-
                     01C6   145
FFFFFFFF             025B   146                      .LONG       ^XFFFFFFFF      ; PSEUDO-UIC
00000000             025F   147                      .LONG       0               ; PSEUDO-UIC
00000267             0263   148                      .BLKL       1               ; UIC
0000026B             0267   149                      .BLKL       1               ; UIC
0000026F             026B   150                      .BLKL       1               ; UIC
                     026F   151 ;
                     026F   152          COND    4,NOTARG,<ORDERING OF CANCEL/WAKE/REPEAT>,-
                     026F   153                    <CANCEL, WAKE, REPEAT>,-
                     026F   154                    <WAKE, CANCEL, REPEAT>,-
                     026F   155                    <WAKE, REPEAT, CANCEL>,-
                     026F   156                    <WAKE, CANCEL>,-
                     026F   157
00000090'00000120'   02EB   158                      .ADDRESS    ABS_3SEC,DELTA_1SEC   ; DAYTIM, REPTIM ARG ADDRESSES
000000A0'00000090'   02F3   159                      .ADDRESS    DELTA_1SEC,DELTA_3SEC ; DAYTIM, REPTIM ARG ADDRESSES
00000000'00000000'   02FB   160                      .ADDRESS    ONES,ONES             ; DAYTIM, REPTIM ARG ADDRESSES
00000000'00000090'   0303   161                      .ADDRESS    DELTA_1SEC,0          ; DAYTIM, REPTIM ARG ADDRESSES
                     030B   162 ;
                     030B   163          COND    5,NULL
                     030C   164
                 00000000   165          .PSECT  SATSSS61,RD,WRT,EXE
```

```
                                                H 10
0000   167          .SBTTL  TM_SETUP, TM_CLEANUP
0000   168 ;++
0000   169 ; FUNCTIONAL DESCRIPTION:
0000   170 ;
0000   171 ;              TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
0000   172 ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
0000   173 ; TEST MODULE EXECUTION.
0000   174 ;
0000   175 ; CALLING SEQUENCE:
0000   176 ;
0000   177 ;       BSBW TM_SETUP   BSBW TM_CLEANUP
0000   178 ;
0000   179 ; INPUT PARAMETERS:
0000   180 ;
0000   181 ;       NONE
0000   182 ;
0000   183 ; IMPLICIT INPUTS:
0000   184 ;
0000   185 ;       NONE
0000   186 ;
0000   187 ; OUTPUT PARAMETERS:
0000   188 ;
0000   189 ;       NONE
0000   190 ;
0000   191 ; IMPLICIT OUTPUTS:
0000   192 ;
0000   193 ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
0000   194 ;                  ALL PRIVILEGES ACQUIRED.
0000   195 ;
0000   196 ; COMPLETION CODES:
0000   197 ;
0000   198 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
0000   199 ;
0000   200 ; SIDE EFFECTS:
0000   201 ;
0000   202 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
0000   203 ;       (VIA RSB) IF ERROR ENCOUNTERED.
0000   204 ;
0000   205 ;--
0000   206
0000   207
0000   208
0000   209 TM_SETUP::
                52   D4  0000   210          CLRL    R2                          ; INITIALIZE
                53   D4  0002   211          CLRL    R3                          ; .. CONDITION
                54   D4  0004   212          CLRL    R4                          ; .... TABLE
                55   D4  0006   213          CLRL    R5                          ; ...... INDEX
                56   D4  0008   214          CLRL    R6                          ; ........ REGISTERS
            FFF3'    30  000A   215          BSBW    MOD_MSG_PRINT               ; PRINT TEST MODULE BEGIN MSG
00000000'EF 00000000'EF  DE  000D   216          MOVAL   TEST_MOD_SUCC,TMD_ADDR  ; ASSUME END MSG WILL SHOW SUCCESS
   03   00 00000000'8F   F0  0018   217          INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
           00000000'EF          0020
                        0025   218          MODE    TO,S$,KRNL                  ; KERNEL MODE TO ACCESS PHD
        59 00000000'9F   D0  0048   219          MOVL    @#CTL$GL_PHD,R9             ; GET PROCESS HEADER ADDRESS
        00000000'EF   69 DE  004F   220          MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                        0056   221          MODE    FROM,S$ ; BACK TO USER MODE
                        0057   222          PRIV    ADD,ALL                     ; GET ALL PRIVILEGES
```

```
                     0077   223          $SETPRN_S TEST_MOD_NAME_D          ; SET PROCESS NAME
                     0084   224          SS_CHECK  NORMAL                   ; CHECK STATUS CODE RETURNED FROM SETPRN
                     00B2   225          $WAKE_S SELFPID                    ; GET MY PID
                     00C1   226          SS_CHECK NORMAL                    ; CHECK FOR NORMAL RETURN
                     00EF   227          $HIBER_S                           ; UNDO ABOVE WAKE
                     00F6   228          SS_CHECK NORMAL                    ; CHECK FOR NORMAL RETURN
                     0124   229 ;
                     0124   230 ; THE FOLLOWING CODE ESTABLISHES UIC'S IN THE CONDITION 3 TABLE
                     0124   231 ;
                     0124   232          MODE    TO,20$,KRNL                ; KERNEL MODE TO ACCESS PCB
      59  00000000'9F   D0   0147   233          MOVL    @#SCH$GL_CURPCB,R9 ; GET CURRENT PCB ADDRESS
          59  00BC C9   D0   014E   234          MOVL    PCB$L_UIC(R9),R9   ; PICK UP UIC FROM PCB
                     0153   235          MODE    FROM,20$                   ; ... AND GET BACK TO USER MODE
                     0154   236 ;
                     0154   237 ; R9 NOW CONTAINS "MY" UIC
                     0154   238 ;
            5A   02  9A   0154   239          MOVZBL  #2,R10                     ; GET COND3 TABLE INDEX NUMBER INTO A REG
      59  00010000 8F   C1   0157   240          ADDL3   #^X10000,R9,COND3_E[R10] ; PUT DIFF GROUP UIC INTO 3RD TABLE ELT
        0000025B'EF4A
                     015E
            5A   D6   0164   241          INCL    R10                        ; POINT TO 4TH COND3 TABLE ELEMENT
 0000025B'EF4A   59   D0   0166   242          MOVL    R9,COND3_E[R10]            ; PUT MY UIC INTO TABLE
            5A   D6   016E   243          INCL    R10                        ; POINT TO 5TH COND3 TABLE ELEMENT
0000025B'EF4A   59   01   C1   0170   244          ADDL3   #1,R9,COND3_E[R10]         ; PUT DIFF MEMBER UIC INTO THE TABLE
                     0179   245          $CREMBX_S CHAN=MBXCHAN, LOGNAM=SUBJPRN, - ; GET MAILBOX FOR PROCESS
                     0179   246                    MAXMSG=#120, PROMSK=#0, BUFQUO=#240
                     019E   247          SS_CHECK NORMAL                    ; CHECK NORMAL COMPLETION
                     01CC   248          $GETCHN_S CHAN=MBXCHAN, -          ; GET CHAN INFO (UNIT NUMBER)
                     01CC   249                    PRIBUF=MBXCHANINFO
                     01E6   250          SS_CHECK NORMAL                    ; CHECK NORMAL COMPLETION
00000088'EF  00000020'EF   3C   0214   251          MOVZWL  MBXCHANINFO+8+DIB$W_UNIT,MBXUNIT ; SAVE MAILBOX UNIT NUMBER
                     021F   252          $BINTIM_S TIMBUF=TIME_PAST, -      ; SET UP A PAST TIME IN ABSOLUTE FORMAT
                     021F   253                    TIMADR=ABS_PAST
                  05 0232   254          RSB                                ; RETURN TO MAIN ROUTINE
                     0233   255 TM_CLEANUP::
                     0233   256          $DELMBX_S  MBXCHAN                 ; DELETE TERMINATION MAILBOX
                FDBC' 30 0241   257          BSBW    MOD_MSG_PRINT              ; PRINT TEST MODULE END MSG
                  05 0244   258          RSB                                ; RETURN TO MAIN ROUTINE
```

```
                        0245   260          .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
                        0245   261 ;++
                        0245   262 ; FUNCTIONAL DESCRIPTION:
                        0245   263 ;
                        0245   264 ;                   CONDX AND CONDX_CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
                        0245   265 ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
                        0245   266 ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
                        0245   267 ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
                        0245   268 ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
                        0245   269 ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
                        0245   270 ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
                        0245   271 ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
                        0245   272 ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
                        0245   273 ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
                        0245   274 ;
                        0245   275 ; CALLING SEQUENCE:
                        0245   276 ;
                        0245   277 ;       BSBW CONDX   BSBW CONDX_CLEANUP
                        0245   278 ;         WHERE X = 1,2,3,4,5
                        0245   279 ;
                        0245   280 ; INPUT PARAMETERS:
                        0245   281 ;
                        0245   282 ;       CONFLICT = 0
                        0245   283 ;
                        0245   284 ; IMPLICIT INPUTS:
                        0245   285 ;
                        0245   286 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                        0245   287 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                        0245   288 ;
                        0245   289 ; OUTPUT PARAMETERS:
                        0245   290 ;
                        0245   291 ;       CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
                        0245   292 ;
                        0245   293 ; IMPLICIT OUTPUTS:
                        0245   294 ;
                        0245   295 ;       R2,3,4,5,6 PRESERVED
                        0245   296 ;
                        0245   297 ; COMPLETION CODES:
                        0245   298 ;
                        0245   299 ;       NONE
                        0245   300 ;
                        0245   301 ; SIDE EFFECTS:
                        0245   302 ;
                        0245   303 ;       NONE
                        0245   304 ;
                        0245   305 ;--
                        0245   306
                        0245   307
                        0245   308
                        0245   309 COND1::
 05                     0245   310          RSB                             ; RETURN TO MAIN ROUTINE
                        0246   311 COND1_CLEANUP::
 05                     0246   312          RSB                             ; RETURN TO MAIN ROUTINE
                        0247   313 COND2::
 05                     0247   314          RSB                             ; RETURN TO MAIN ROUTINE
                        0248   315 COND2_CLEANUP::
 05                     0248   316          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                     0249   317 COND3::
                02   54  D1 0249   318         CMPL    R4,#2                   ; DOES CONDITION 3 SPECIFY DIFFERENT GROUP ?
                     35  13 024C   319         BEQL    20$                     ; YES -- THIS IS CONFLICT BECAUSE OF
                     024E   320                                                 ; ... USE OF COMMON CLUSTERS
0000017C'EF42   0000011C'8F  D1 024E   321         CMPL    #SUBJPID,COND1_E[R2]    ; NON-ZERO PID SPECIFIED ?
                     19  13 025A   322         BEQLU   10$                     ; YES -- PROCESS IS "OTHER"
              000001BE'EF43  D5 025C   323         TSTL    COND2_E[R3]             ; IS PROCESS NAME SPECIFIED ?
                     10  12 0263   324         BNEQ    10$                     ; YES -- SUBJECT PROCESS IS "OTHER"
                     0265   325 5$:
                     0265   326 ;
                     0265   327 ; PROCESS IS "SELF"
                     0265   328 ;
0000025B'EF44   00000000'EF  D1 0265   329         CMPL    ONES,COND3_E[R4]        ; DOES CONDITION 3 SPECIFY "SELF" ?
                     1B  13 0271   330         BEQLU   COND3X                  ; YES -- THEN ALL 3 CONDIT'NS ARE CONSISTENT
                     0E  11 0273   331         BRB     20$                     ; NO -- INDICATE CONFLICT & GET OUT
                     0275   332 10$:
                     0275   333 ;
                     0275   334 ; PROCESS IS "OTHER"
                     0275   335 ;
0000025B'EF44   00000000'EF  D1 0275   336         CMPL    ONES,COND3_E[R4]        ; DOES CONDITION 3 SPECIFY "SELF" ?
                     0B  12 0281   337         BNEQU   COND3X                  ; NO -- THEN ALL 3 CONDITIONS ARE CONSISTENT
                     0283   338 20$:
  00000000'EF   00000000'EF  90 0283   339         MOVB    ONES,CONFLICT           ; YES -- INDICATE CONFLICT
                     028E   340 COND3X:
                     05 028E   341         RSB                             ; RETURN TO MAIN ROUTINE
                     028F   342 COND3_CLEANUP::
                     05 028F   343         RSB                             ; RETURN TO MAIN ROUTINE
                     0290   344 COND4::
                     05 0290   345         RSB                             ; RETURN TO MAIN ROUTINE
                     0291   346 COND4_CLEANUP::
                     05 0291   347         RSB                             ; RETURN TO MAIN ROUTINE
                     0292   348 COND5::
                     05 0292   349         RSB                             ; RETURN TO MAIN ROUTINE
                     0293   350 COND5_CLEANUP::
                     05 0293   351         RSB                             ; RETURN TO MAIN ROUTINE
```

```
                     0294    353          .SBTTL  FORM_CONDS
                     0294    354 ;++
                     0294    355 ; FUNCTIONAL DESCRIPTION:
                     0294    356 ;
                     0294    357 ;               FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
                     0294    358 ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
                     0294    359 ;
                     0294    360 ; CALLING SEQUENCE:
                     0294    361 ;
                     0294    362 ;       BSBW FORM_CONDS
                     0294    363 ;
                     0294    364 ; INPUT PARAMETERS:
                     0294    365 ;
                     0294    366 ;       NONE
                     0294    367 ;
                     0294    368 ; IMPLICIT INPUTS:
                     0294    369 ;
                     0294    370 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                     0294    371 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                     0294    372 ;       FOR X = 1,2,3,4,5 :
                     0294    373 ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
                     0294    374 ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
                     0294    375 ;               CONDX_C - CONTEXT OF THE CONDX TABLE
                     0294    376 ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
                     0294    377 ;
                     0294    378 ; OUTPUT PARAMETERS:
                     0294    379 ;
                     0294    380 ;       NONE
                     0294    381 ;
                     0294    382 ; IMPLICIT OUTPUTS:
                     0294    383 ;
                     0294    384 ;       NONE
                     0294    385 ;
                     0294    386 ; COMPLETION CODES:
                     0294    387 ;
                     0294    388 ;       NONE
                     0294    389 ;
                     0294    390 ; SIDE EFFECTS:
                     0294    391 ;
                     0294    392 ;       NONE
                     0294    393 ;
                     0294    394 ;--
                     0294    395
                     0294    396
                     0294    397
                     0294    398 FORM_CONDS::
                     0294    399          $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
                     02B3    400                                          ; FORMAT CONDITIONS HEADER MSG
                FD4A'  30    02B3    401          BSBW    OUTPUT_MSG      ; ... AND PRINT IT
              14   00  91    02B6    402          CMPB    #COND1_C,#NULL  ; IS CONDITION 1 NULL ?
                   03  12    02B9    403          BNEQU   10$             ; NO -- CONTINUE
                00BF   31    02BB    404          BRW     FORM_CONDSX     ; YES -- SUBROUTINE IS FINISHED
                     02BE    405 10$:
  00000000'EF   00000131'EF  DE    02BE    406          MOVAL   COND1_T,MSG_A           ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF   0000013E'EF42  D0    02C9    407          MOVL    COND1_TAB[R2],MSG_B     ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
         00000000'EF   00    90    02D5    408          MOVB    #COND1_C,MSG_CTXT       ; SAVE CONDITION 1 CONTEXT FOR FAO
                     02DC    409          MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                FD21'  30  02DC   410          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 1 MSG
              14   00  91  02DF   411          CMPB    #COND2_C,#NULL              ; IS CONDITION 2 NULL ?
                   03  12  02E2   412          BNEQU   20$                         ; NO -- CONTINUE
                0096  31  02E4   413          BRW     FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
                      02E7   414 20$:
  00000000'EF   00000188'EF  DE  02E7   415          MOVAL   COND2_T,MSG_A               ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF   0000019E'EF43  D0  02F2   416          MOVL    COND2_TAB[R3],MSG_B         ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
        00000000'EF   00  90  02FE   417          MOVB    #COND2_C,MSG_CTXT           ; SAVE CONDITION 2 CONTEXT FOR FAO
                      0305   418          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                FCF8'  30  0305   419          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 2 MSG
              14   00  91  0308   420          CMPB    #COND3_C,#NULL              ; IS CONDITION 3 NULL ?
                   03  12  030B   421          BNEQU   30$                         ; NO -- CONTINUE
                006D  31  030D   422          BRW     FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
                      0310   423 30$:
  00000000'EF   000001C6'EF  DE  0310   424          MOVAL   COND3_T,MSG_A               ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF   000001D4'EF44  D0  031B   425          MOVL    COND3_TAB[R4],MSG_B         ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
        00000000'EF   00  90  0327   426          MOVB    #COND3_C,MSG_CTXT           ; SAVE CONDITION 3 CONTEXT FOR FAO
                      032E   427          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                FCCF'  30  032E   428          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 3 MSG
              14   00  91  0331   429          CMPB    #COND4_C,#NULL              ; IS CONDITION 4 NULL ?
                   47  13  0334   430          BEQLU   FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   0000026F'EF  DE  0336   431          MOVAL   COND4_T,MSG_A               ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF   0000028F'EF45  D0  0341   432          MOVL    COND4_TAB[R5],MSG_B         ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
        00000000'EF   00  90  034D   433          MOVB    #COND4_C,MSG_CTXT           ; SAVE CONDITION 4 CONTEXT FOR FAO
                      0354   434          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                FCA9'  30  0354   435          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 4 MSG
              14   14  91  0357   436          CMPB    #COND5_C,#NULL              ; IS CONDITION 5 NULL ?
                   21  13  035A   437          BEQLU   FORM_CONDSX                 ; YES -- SUBROUTINE IS FINISHED
  00000000'EF   0000030B'EF  DE  035C   438          MOVAL   COND5_T,MSG_A               ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF   0000030B'EF46  D0  0367   439          MOVL    COND5_TAB[R6],MSG_B         ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
        00000000'EF   14  90  0373   440          MOVB    #COND5_C,MSG_CTXT           ; SAVE CONDITION 5 CONTEXT FOR FAO
                      037A   441          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                FC83'  30  037A   442          BSBW    WRITE_MSG2                  ; FORMAT AND WRITE CONDITION 5 MSG
                      037D   443 FORM_CONDSX:
                   05  037D   444          RSB                                 ; RETURN TO CALLER
```

```
                        .SBTTL  VERIFY
                ;++
                ; FUNCTIONAL DESCRIPTION:
                ;
                ;               VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
                ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
                ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
                ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
                ; ($SCH/CANWAK). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
                ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
                ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
                ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
                ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
                ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
                ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
                ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
                ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
                ;
                ; CALLING SEQUENCE:
                ;
                ;       BSBW VERIFY
                ;
                ; 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:
                ;
                ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
                ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
                ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
                ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
                ;       ERRORS.
                ;
                ; 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.
                ;
```

```
                                037E   503 ;--
                                037E   504
                                037E   505
                                037E   506
                                037E   507 VERIFY::
              00000000'EF   95  037E   508         TSTB    CFLAG                   ; SHOULD CONDITIONS BE PRINTED ?
                      03    13  0384   509         BEQL    5$                      ; NO -- CONTINUE
                    FF0B    30  0386   510         BSBW    FORM_CONDS              ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                                0389   511 5$:
    0000011C'EF   00000114'EF   D0  0389   512         MOVL    SELFPID,SUBJPID         ; ASSUME THE SUBJECT PID IS SELF
              00000110'EF       D4  0394   513         CLRL    ZEROPID                 ; CLEAR ZERO PID
              00000130'EF       94  039A   514         CLRB    LONG_WAIT               ; INITIALIZE LONG WAIT INDICATOR
 0000025B'EF44  00000000'EF       D1  03A0   515         CMPL    ONES,COND3_E[R4]        ; IS PROCESS FOR THIS TEST CASE SELF ?
                      03    12  03AC   516         BNEQU   7$                      ; NO -- CONTINUE
                    0070    31  03AE   517         BRW     10$                     ; YES -- DON'T CREATE A PROCESS
                                03B1   518 7$:
                                03B1   519         $CREPRC_S PIDADR=CREPID, PRCNAM=SUBJPRN, -
                                03B1   520                   UIC=COND3_E[R4], IMAGE=IMAGNAM, -
                                03B1   521                   MBXUNT=MBXUNIT,, QUOTA=QUOTALIST
                                03E8   522                                           ; CREATE THE SUBJECT PROCESS
                                03E8   523         SS_CHECK NORMAL                   ; ... AND MAKE SURE IT CREATED OK
    0000011C'EF   00000118'EF   D0  0416   524         MOVL    CREPID,SUBJPID          ; MAKE THE SUBJCT PID = THE ONE JUST CREATED
                                0421   525 10$:
0000010C'EF   0000017C'EF42     D0  0421   526         MOVL    COND1_E[R2],DEST_PIDADR ; GET PID ADDRESS OUT OF TABLE
          59  000001BE'EF43     D0  042D   527         MOVL    COND2_E[R3],R9          ; PRCNAM ADDR INTO REG FOR INDIRECT REF'RNCE
          57  000002EB'EF45     7D  0435   528         MOVQ    COND4_E[R5],R7          ; GET DAYTIM, REPTIM ARG ADDRESSES INTO REGS
                                043D   529         $CANWAK_S SUBJPID                 ; ISSUE PRELIM CANWAK TO CLEAR THE DECKS
                                044C   530         SS_CHECK NORMAL                   ; CHECK FOR NORMAL RETURN
                                047A   531         $SETIMR_S DAYTIM=DELTA_10SEC, -   ; SET "WATCHDOG" TIMER TO TRIP IF LONG WAIT
                                047A   532                   ASTADR=WATCH_AST
                                0491   533         SS_CHECK NORMAL                   ; CHECK FOR NORMAL RETURN
                                04BF   534         $GETTIM_S ABS_3SEC                ; GET CURRENT TIME
                                04CC   535         SS_CHECK NORMAL                   ; CHECK FOR NORMAL RETURN
00000120'EF   000000B8'EF       C0  04FA   536         ADDL    POS_3SEC,ABS_3SEC       ; ADD 3 SECONDS TO LOWER LONGWORD
00000124'EF   000000BC'EF       D8  0505   537         ADWC    POS_3SEC+4,ABS_3SEC+4   ; ADD POSSIBLE CARRY TO HIGHER LONGWORD
                                0510   538                                           ; ABS_3SEC IS NOW VALID IF USED IN $SCHDWK
                                0510   539 ;
                                0510   540 ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                                0510   541 ;
                                0510   542         $SCHDWK_S PIDADR=@DEST_PIDADR, PRCNAM=(R9), -
                                0510   543                   DAYTIM=(R7), REPTIM=(R8)
          00000000'8F   50  D1  0523   544         CMPL    R0,#SS$_NORMAL          ; CODE RECEIVED = CODE EXPECTED ?
                        61  13  052A   545         BEQLU   15$                     ; YES -- CONTINUE
00000000'EF   00000000'8F   D0  052C   546         MOVL    #SS$_NORMAL,EXPV        ; NO -- LOAD UP EXPECTED AND ...
          00000000'EF   50  D0  0537   547         MOVL    R0,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                                053E   548         ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM SCHDWK>
                                058D   549 15$:
              0000010C'EF   D5  058D   550         TSTL    DEST_PIDADR             ; PID RETURNED BY SCHDWK ?
                        68  13  0593   551         BEQL    20$                     ; NO -- KEEP GOING
0000010C'FF   0000011C'EF   D1  0595   552         CMPL    SUBJPID,@DEST_PIDADR    ; YES -- IS IT THE CORRECT ONE ?
                        5B  13  05A0   553         BEQL    20$                     ; YES -- CONTINUE
00000000'EF   0000011C'EF   D0  05A2   554         MOVL    SUBJPID,EXPV            ; NO --LOAD UP EXPECTED AND
00000000'EF   0000010C'FF   D0  05AD   555         MOVL    @DEST_PIDADR,RECV       ; ... RECEIVED VALUES, THEN EXIT
                                05B8   556         ERR_EXIT LONG,<INCORRECT PID RETURNED BY SCHDWK>
                                05FD   557 20$:
                                05FD   558         $CLREF_S EFN=#32                ; CLEAR EVENT FLAG 32
                     2E 50      E8  0606   559         BLBS    R0,25$                  ; KEEP GOING IF OK
```

```
                                0609   560          SS_CHECK NORMAL                 ; USE SS_CHECK MACRO TO TERMINATE TEST MOD
                                0637   561 25$:
                                0637   562          $SETIMR_S EFN=#32, -            ; SET A 2-SECOND TIMER
                                0637   563                    DAYTIM=DELTA_2SEC
                                0648   564          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                                0676   565          $WAITFR_S EFN=#32               ; WAIT 2 SECONDS TO ALLOW PROPER SYNCH'N
                                067F   566          SS_CHECK NORMAL                 ; CHECK FOR NORMAL RETURN
                00000110'EF   D4 06AD   567          CLRL    ZEROPID                 ; CLEAR OUT ZERO PID SCHDWK MAY HAVE SET
                                06B3   568 ;
                                06B3   569 ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                                06B3   570 ;
                                06B3   571          $CANWAK_S PIDADR=@DEST_PIDADR, PRCNAM=(R9)
                                06C2   572                                          ; CANCEL SCHEDULED WAKE OR REPEAT
            00000000'8F   50    D1 06C2   573          CMPL    R0,#SS$_NORMAL          ; CODE RECEIVED = CODE EXPECTED ?
                          61    13 06C9   574          BEQLU   30$                     ; YES -- CONTINUE
  00000000'EF   00000000'8F    D0 06CB   575          MOVL    #SS$_NORMAL,EXPV        ; NO -- LOAD UP EXPECTED AND ...
            00000000'EF   50    D0 06D6   576          MOVL    R0,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                                06DD   577          ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM CANWAK>
                                072C   578 30$:
                00000010C'EF    D5 072C   579          TSTL    DEST_PIDADR             ; PID RETURNED BY CANWAK ?
                          68    13 0732   580          BEQL    40$                     ; NO -- KEEP GOING
  0000010C'FF   0000011C'EF    D1 0734   581          CMPL    SUBJPID,@DEST_PIDADR    ; YES -- IS IT THE CORRECT ONE ?
                          5B    13 073F   582          BEQL    40$                     ; YES -- CONTINUE
  00000000'EF   0000011C'EF    D0 0741   583          MOVL    SUBJPID,EXPV            ; NO --LOAD UP EXPECTED AND
  00000000'EF   0000010C'FF    D0 074C   584          MOVL    @DEST_PIDADR,RECV       ; ... RECEIVED VALUES, THEN EXIT
                                0757   585          ERR_EXIT LONG,<INCORRECT PID RETURNED BY CANWAK>
                                079C   586 40$:
  0000011C'EF   00000118'EF    D1 079C   587          CMPL    CREPID,SUBJPID          ; WAS A PROCESS CREATED ?
                          03    13 07A7   588          BEQLU   50$                     ; YES -- GO WAIT FOR IT TO END
                      014F    31 07A9   589          BRW     60$                     ; NO -- GO ISSUE HIBER
                                07AC   590 50$:
                                07AC   591          $ASCEFC_S EFN=#64, NAME=CLUSTER ; ASSOC WITH CLUSTER FOR PROCESS SYNCHRO'N
                                07C3   592          SS_CHECK  NORMAL                ; CHECK FOR NORMAL STATUS
                                07F1   593          $SETEF_S  EFN=#65               ; LET CREATED PROC EXIT
                                07FE   594          SS_CHECK  WASCLR                ; BIT 65 SHOULD HAVE BEEN CLEAR
                                082C   595          $WAITFR_S EFN=#64               ; WAIT UNTIL CREATED PROC CAN HIBERNATE
                                0839   596          SS_CHECK  NORMAL                ; CHECK FOR NORMAL RETURN
                                0867   597          $DACEFC_S EFN=#64               ; DISASSOC CLUSTER
                                0874   598          SS_CHECK  NORMAL                ; CHECK FOR NORMAL STATUS
                                08A2   599          $QIOW_S CHAN=MBXCHAN, FUNC=#IO$_READVBLK, -
                                08A2   600                  P1=MBXBUFF+8, P2=MBXBUFF
                                08CB   601                                          ; WAIT FOR CREATED PROCESS TO SEND MAIL
                                08CB   602          SS_CHECK NORMAL                 ; CHECK FOR NORMAL STATUS CODE
                          35    11 08F9   603          BRB     70$                     ; ... AND GO SEE IF WE WERE STUCK IN HIBER
                                08FB   604 60$:
                                08FB   605          $HIBER_S                        ; HIBERNATE TO SATISFY OUTSTANDING WAKE
                                0902   606          SS_CHECK NORMAL                 ; ... MAKE SURE IT FINISHED OK
                                0930   607 70$:
                                0930   608 ;
                                0930   609 ;  CHECK TO SEE IF STUCK IN HIBER ..... IF LONG WAIT
                                0930   610 ;  IS SET AND DID NOT EXPECT LONG WAIT, ISSUE ERR_EXIT
                                0930   611 ;  SAYING "STUCK IN HIBER".
                                0930   612 ;
    00000130'EF   95    0930   613          TSTB    LONG_WAIT               ; DID WE WAIT A LONG TIME ?
                  5C    13 0936   614          BEQL    VERIFYX                 ; NO -- THIS TEST CASE IS FINISHED
                  55    D5 0938   615          TSTL    R5                      ; YES -- DID WE EXPECT TO REMAIN IN HIB'N ?
                  58    13 093A   616          BEQL    VERIFYX                 ; YES -- THAT'S OK
```

```
                                                D 11
00000000'EF   94  093C   617          CLRB    EXPV                    ; NO -- SOMETHING WENT WRONG .... LOAD UP
00000000'EF   94  0942   618          CLRB    RECV                    ; ... EXPECTED & RECEIVED VALUES, THEN EXIT
                  0948   619          ERR_EXIT BYTE,<SUBJECT PROCESS WAS LEFT IN HIBERNATION>
                  0994   620 VERIFYX:
                  05 0994   621          RSB                             ; RETURN TO CALLER
```

```
0995   623          .SBTTL  VFY_CLEANUP
0995   624 ;++
0995   625 ; FUNCTIONAL DESCRIPTION:
0995   626 ;
0995   627 ;               VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
0995   628 ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
0995   629 ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
0995   630 ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
0995   631 ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
0995   632 ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
0995   633 ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
0995   634 ; POSSIBLY DISCOVERING A SECOND ERROR.
0995   635 ;
0995   636 ; CALLING SEQUENCE:
0995   637 ;
0995   638 ;       BSBW VFY_CLEANUP
0995   639 ;
0995   640 ; INPUT PARAMETERS:
0995   641 ;
0995   642 ;       NONE
0995   643 ;
0995   644 ; IMPLICIT INPUTS:
0995   645 ;
0995   646 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
0995   647 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
0995   648 ;       FOR X = 1,2,3,4,5 :
0995   649 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
0995   650 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
0995   651 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
0995   652 ;                 FOR CONDX_E.
0995   653 ;
0995   654 ; OUTPUT PARAMETERS:
0995   655 ;
0995   656 ;       NONE
0995   657 ;
0995   658 ; IMPLICIT OUTPUTS:
0995   659 ;
0995   660 ;       NONE
0995   661 ;
0995   662 ; COMPLETION CODES:
0995   663 ;
0995   664 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
0995   665 ;
```

```
                        0995   667 ; SIDE EFFECTS:
                        0995   668 ;
                        0995   669 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                        0995   670 ;       (VIA RSB) IF ERROR ENCOUNTERED.
                        0995   671 ;
                        0995   672 ;--
                        0995   673
                        0995   674
                        0995   675
                        0995   676 VFY_CLEANUP::
                        0995   677         $CANWAK_S SUBJPID                   ; CANCEL ANY POSSIBLE OUTSTANDING WAKES
                        09A4   678         $CANTIM_S                           ; CANCEL WATCHDOG TIMER
0000011C'EF   00000118'EF D1   09AD   679         CMPL    CREPID,SUBJPID      ; WAS A PROCESS CREATED FOR THIS TEST CASE ?
                       0F 12   09B8   680         BNEQU   VFY_CLEANUPX        ; NO -- JUST EXIT
                          09BA 681         $DELPRC_S SUBJPID                   ; YES -- DELETE IT
                          09C9 682 VFY_CLEANUPX:
                       05 09C9   683         RSB                             ; RETURN TO CALLER
```

```
                                09CA   685          .SBTTL  WATCH_AST
                                09CA   686 ;
                                09CA   687 ; WATCH_AST SHOULD BE ENTERED ONLY WHEN THE CREATING OR CREATED
                                09CA   688 ; PROCESS IS HIBERNATING. IT IS SCHEDULED WITH A 10-SECOND TIMER,
                                09CA   689 ; WHICH IS CANCELED BEFORE DELIVERY IN ALL CASES EXCEPT WHEN THE
                                09CA   690 ; SUBJECT PROCESS GOES INTO AN UNSATISFIED HIBERNATION. WHEN
                                09CA   691 ; WATCH_AST IS ENTERED, IT SETS A FLAG INDICATING IT WAS ENTERED
                                09CA   692 ; (LONG_WAIT) TO NON-ZERO, AND ISSUES A $WAKE FOR THE SUBJECT
                                09CA   693 ; PROCESS; THIS SHOULD CLEAR THE HIBERNATION. BACK IN THE MAIN
                                09CA   694 ; ROUTINE, A CHECK IS MADE TO SEE IF THE WATCH_AST WAS ENTERED
                                09CA   695 ; AND WHETHER OR NOT SUCH ENTRY WAS EXPECTED. AN UNEXPECTED ENTRY
                                09CA   696 ; TO WATCH_AST CAUSES AN ERR_EXIT.
                                09CA   697 ;
                                09CA   698 WATCH_AST:
                           0000 09CA   699          .WORD   0                       ; ENTRY MASK
00000130'EF   00000000'EF   90 09CC   700          MOVB    ONES,LONG_WAIT          ; INDICATE THAT THE AST WAS ENTERED
                                09D7   701          $WAKE_S SUBJPID                 ; WAKE THE (PRESUMABLY) HIBERNATING PROCESS
                             04 09E6   702          RET                             ; ... AND GET OUT
                                09E7   703          .END
```

```
$$$$                    = 00000952 R     04
$$$CHARS                = 00000027
$$$CHARS1               = 00000014
$$$CHARS2               = 00000014
$$$CHARS3               = 00000014
$$$CHARS4               = 0000000C
$$$CHARS5               = 00000000
$$$COND_A               = 00000003
$$$STRINGS              = 00000001
$$$STRINGS2             = 00000005
$$T1                    = 00000001
$$T2                    = 00000004
ABS_3SEC                  00000120 R     03
ABS_PAST                  00000128 R     03
BYTE                    = 00000001   G
CFLAG                     ********     X 04
CHMRTN                    ********     X 04
CHM_CONT                  ********     X 04
CLUSTER                   00000084 R     02
COMP_SC                   ********   X   04
CONDT                     00000245 RG    04
COND1_C                 = 00000000
COND1_CLEANUP             00000246 RG    04
COND1_E                   0000017C R     03
COND1_H                   0000013D RG    03
COND1_T                   00000131 R     03
COND1_TAB                 0000013E R     03
COND2                     00000247 RG    04
COND2_C                 = 00000000
COND2_CLEANUP             00000248 RG    04
COND2_E                   000001BE R     03
COND2_H                   0000019D RG    03
COND2_T                   00000188 R     03
COND2_TAB                 0000019E R     03
COND3                     00000249 RG    04
COND3X                    0000028E R     04
COND3_C                 = 00000000
COND3_CLEANUP             0000028F RG    04
COND3_E                   0000025B R     03
COND3_H                   000001D3 RG    03
COND3_T                   000001C6 R     03
COND3_TAB                 000001D4 R     03
COND4                     00000290 RG    04
COND4_C                 = 00000000
COND4_CLEANUP             00000291 RG    04
COND4_E                   000002EB R     03
COND4_H                   0000028E RG    03
COND4_T                   0000026F R     03
COND4_TAB                 0000028F R     03
COND5                     00000292 RG    04
COND5_C                 = 00000014
COND5_CLEANUP             00000293 RG    04
COND5_H                   0000030B RG    03
COND5_T                   0000030B R     03
COND5_TAB                 0000030B R     03
CONFLICT                  ********   X   04
CREPID                    00000118 R     03
CTL$GL_PHD                ********     X 04
DELTA_TOSEC               000000A8 R     02
DELTA_1SEC                00000090 R     02
DELTA_2SEC                00000098 R     02
DELTA_3SEC                000000A0 R     02
DELTA_QSEC                000000B0 R     02
DESC                    = 00000010   G
DEST_PIDADR               0000010C R     03
DIB$R_LENGTH            = 00000074
DIB$W_UNIT              = 0000000C
EFLAG                     ********     X 04
EXPV                      ********     X 04
FAO_DESC                  ********     X 04
FAO_LEN                   ********     X 04
FORM_CONDS                00000294 RG    04
FORM_CONDSX               0000037D R     04
IMAGNAM                   00000065 R     02
IO$_READVBLK              ********     X 04
LONG                    = 00000004   G
LONG_WAIT                 00000130 R     03
MBXBOFF                   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
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 03
ONE_SEC                 = 00989680
OUTPUT_MSG                ********     X 04
PCB$L_UIC               = 000000BC
PCV                       ********     X 04
PHD$Q_PRIVMSK           = 00000000
POS_3SEC                  000000B8 R     02
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$GL_CURPCB             ********     X 04
SELFPID                   00000114 R     03
SS$_NORMAL                ********     X 04
SS$_WASCLR                ********     X 04
SUBJPID                   0000011C R     03
SUBJPRN                   00000051 R     02
SUCCESS                   ********     X 04
SYS$ASCEFC                ********   GX  04
SYS$BINTIM                ********   GX  04
SYS$CANTIM                ********   GX  04
```

```
SYS$CANWAK                      ********  GX    04
SYS$CLREF                       ********  GX    04
SYS$CMKRNL                      ********  GX    04
SYS$CREMBX                      ********  GX    04
SYS$CREPRC                      ********  GX    04
SYS$DACEFC                      ********  GX    04
SYS$DELMBX                      ********  GX    04
SYS$DELPRC                      ********  GX    04
SYS$FAO                         ********   X    04
SYS$GETCHN                      ********  GX    04
SYS$GETTIM                      ********  GX    04
SYS$HIBER                       ********  GX    04
SYS$QIOW                        ********  GX    04
SYS$SCHDWK                      ********  GX    04
SYS$SETEF                       ********  GX    04
SYS$SETIMR                      ********  GX    04
SYS$SETPRN                      ********  GX    04
SYS$SETPRV                      ********  GX    04
SYS$WAITFR                      ********  GX    04
SYS$WAKE                        ********  GX    04
TESTNUM                         ********   X    04
TEST_MOD_NAME                   00000000 RG     02
TEST_MOD_NAME_D                 00000009 R      02
TEST_MOD_SUCC                   ********     X  04
TIME_PAST                       000000C0 R      02
TMD_ADDR                        ********     X  04
TM_CLEANUP                      00000233 RG     04
TM_SETUP                        00000000 RG     04
VERIFY                          0000037E RG     04
VERIFYX                         00000994 R      04
VFY_CLEANUP                     00000995 RG     04
VFY_CLEANUPX                    000009C9 R      04
WATCH_AST                       000009CA 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	000000E0 (224.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	0000030C (780.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS61	000009E7 (2535.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.08	00:00:00.36
Command processing	113	00:00:00.63	00:00:02.52

Pass 1	329	00:00:10.90	00:00:18.26
Symbol table sort	0	00:00:00.88	00:00:00.96
Pass 2	160	00:00:02.72	00:00:42.82
Symbol table output	18	00:00:00.12	00:00:00.14
Psect synopsis output	2	00:00:00.02	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	653	00:00:15.35	00:01:05.09

The working set limit was 1500 pages.
58672 bytes (115 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 512 non-local and 76 local symbols.
703 source lines were read in Pass 1, producing 29 object records in Pass 2.
56 pages of virtual memory were used to define 46 macros.

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

Macro library name	Macros defined
_255DUA28:[SHRLIB]UETP.MLB;1	8
_255DUA28:[SYS.OBJ]LIB.MLB;1	2
_255DUA28:[SYSLIB]STARLET.MLB;2	33
TOTALS (all libraries)	43

943 GETS were required to define 43 macros.

There were no errors, warnings or information messages.

```
MACRO/LIS=LIS$:SATSSS61/OBJ=OBJ$:SATSSS61 MSRC$:SATSSS61/UPDATE=(ENH$:SATSSS61)+EXECML$/LIB+SHRLIB$:UETP/LIB
```

SATSSS56
LIS

SATSSS60
LIS

SATSSS71
LIS

SATSSS54
LIS

SATSSS73
LIS

SATSSS70
LIS

SATSSS72
LIS

SATSSS55
LIS

SATSSS53
LIS

SATSSS61
LIS