


```

SSSSSSSS  AAAAAA  TTTTTTTTTT  SSSSSSSS  SSSSSSSS  SSSSSSSS  44  44  5555555555
SSSSSSSS  AAAAAA  TTTTTTTTTT  SSSSSSSS  SSSSSSSS  SSSSSSSS  44  44  5555555555
SS        AA      AA      TT      SS        SS        SS        44  44  55
SS        AA      AA      TT      SS        SS        SS        44  44  55
SS        AA      AA      TT      SS        SS        SS        44  44  555555
SS        AA      AA      TT      SS        SS        SS        44  44  555555
SSSSSSSS  AA      AA      TT      SSSSSSS  SSSSSSS  SSSSSSS  4444444444  55
SSSSSSSS  AA      AA      TT      SSSSSSS  SSSSSSS  SSSSSSS  4444444444  55
SS        AA      AA      TT      SS        SS        SS        44  44  55
SS        AA      AA      TT      SS        SS        SS        44  44  55
SS        AA      AA      TT      SS        SS        SS        44  44  55
SSSSSSSS  AA      AA      TT      SSSSSSSS  SSSSSSSS  SSSSSSSS  44  44  555555
SSSSSSSS  AA      AA      TT      SSSSSSSS  SSSSSSSS  SSSSSSSS  44  44  555555

```

```

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

```

| | | |
|-----|-----|---|
| (1) | 56 | DECLARATIONS |
| (1) | 112 | CONDITION TABLES |
| (1) | 151 | TM_SETUP, TM_CLEANUP |
| (1) | 242 | CONDITION SUBROUTINES - SETUP AND CLEANUP |
| (1) | 335 | FORM_CONDS |
| (1) | 428 | VERIFY |
| (1) | 572 | VFY_CLEANUP |

```

0000 1 .TITLE SATSSS45 SATS SYSTEM SERVICE TESTS $SETPRI (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 SATSSS45 TO TEST SUCCESSFUL
0000 36 : OPERATION OF THE $SETPRI SYSTEM SERVICE. THE SERVICE IS INVOKED
0000 37 : UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000 38 : SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE. CORRECT
0000 39 : OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000 40 : CHECKING FOR AN SS$ 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: NOV, 1977
0000 47
0000 48 : MODIFIED BY:
0000 49
0000 50 : VERSION 1.5 : 25-MAY-79
0000 51
0000 52 : 01 LDJ 10/11/79 Fixed bug caused by DIB$K_LENGTH change ACG052.RNO mem
0000 53
0000 54 :--

```

```
0000 56 .SBTTL DECLARATIONS
0000 57 :
0000 58 : INCLUDE FILES:
0000 59 :
0000 60 $PRVDEF : PRIVILEGE BIT DEFINITIONS
0000 61 $PHDDEF : PROCESS HEADER OFFSETS
0000 62 $PQLDEF : PROCESS QUOTA CODES
0000 63 $PCBDEF : PCB LABELS
0000 64 $DIBDEF : DEVICE INFO BLOCK OFFSETS
0000 65 :
0000 66 : MACROS:
0000 67 :
0000 68 :
0000 69 : EQUATED SYMBOLS:
0000 70 :
0000 71 :
0000 72 : OWN STORAGE:
0000 73 :
```

```

00000000 75 .PSECT RODATA, RD, NOWRT, NOEXE, LONG
0000 76 TEST_MOD_NAME:: STRING C,<SATSSS45> : TEST MODULE NAME
0009 77 TEST_MOD_NAME_D: STRING I,<SATSSS45> : TEST MODULE NAME DESCRIPTOR
0019 78 MSG1_INP_CTL: STRING I,<SSSPR!4ZW: CONDITIONS:>
0039 79 : FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
0039 80 MSG3_ERR_CTL:: STRING I,<*SSSPR!4ZW: !AS>
0051 81 : FAO CTL G FOR MSG3 IN SUCCOMMON.MAR
0051 82 SUBJPRN: STRING I,<SATSSS45 CRE> : PROCESS MBX NAME FOR CREATED PROCESS
0065 83 IMAGNAM: STRING I,<SYSTST$RES:SATSJOB.EXE> : IMAGE NAME FOR CREATED PROC
0084 84 QUOTALIST: $QUOTA CPULM,0 : INFINITE CPU
0089 85 $QUOTA BYTLM,512 : BYTE LIMIT FOR BUFFERED I/O
008E 86 $QUOTA FILLM,2 : OPEN FILE COUNT LIMIT
0093 87 $QUOTA PGFLQUOTA,10 : PAGING FILE QUOTA
0098 88 $QUOTA PRCLM,2 : SUBPROCESS QUOTA
0C9D 89 $QUOTA TQELM,3 : TIMER QUEUE ENTRY QUOTA
00A2 90 $QUOTA LISTEND : DEFINES END OF LIST

```



```

0125 112 .SBTTL CONDITION TABLES
0125 113 :
0125 114 :
0125 115 :
0125 116 :
0125 117 :
0125 118 :
0125 119 :
0125 120 :
00000000' 0170 121 .ADDRESS 0
0000011C' 0174 122 .ADDRESS SUBJPID
00000110' 0178 123 .ADDRESS ZEROPID
017C 124 :
017C 125 :
017C 126 :
017C 127 :
017C 128 :
00000051' 01B2 129 .ADDRESS SUBJPRN
00000000' 01B6 130 .ADDRESS 0
01BA 131 :
01BA 132 :
01BA 133 :
01BA 134 :
01BA 135 :
01BA 136 :
01BA 137 :
01BA 138 :
FFFFFFF 024F 139 .LONG ^XFFFFFFF : PSEUDO-UIC
00000000 0253 140 .LONG 0 : PSEUDO-UIC
000J025B 0257 141 .BLKL 1 : UIC
0000025F 025B 142 .BLKL 1 : UIC
00000263 025F 143 .BLKL 1 : UIC
0263 144 :
0263 145 :
0264 146 :
0264 147 :
0265 148 :
00000000 0265 149 .COND 4, NULL
COND 5, NULL
.PSECT SATSSS45, RD, WRT, EXE

```

SA
SY
SY
SY
SY
SY
SY
TE
TE
TE
TE
TM
TM
TM
VE
VE
VF
VF
WO
WR
ZE

PS
--
: SA
RO
RW
SA

Ph
--
In
Co
Pa
Sy
Pa
Sy
Ps
Cr
As

Th
51
Th
63
47


```

0000 151 .SBTTL TM_SETUP, TM_CLEANUP
0000 152 :++
0000 153 : FUNCTIONAL DESCRIPTION:
0000 154 :
0000 155 : TM SETUP AND TM CLEANUP ARE CALLED TO PERFORM
0000 156 : REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
0000 157 : TEST MODULE EXECUTION.
0000 158 :
0000 159 : CALLING SEQUENCE:
0000 160 :
0000 161 : BSBW TM_SETUP BSBW TM_CLEANUP
0000 162 :
0000 163 : INPUT PARAMETERS:
0000 164 :
0000 165 : NONE
0000 166 :
0000 167 : IMPLICIT INPUTS:
0000 168 :
0000 169 : NONE
0000 170 :
0000 171 : OUTPUT PARAMETERS:
0000 172 :
0000 173 : NONE
0000 174 :
0000 175 : IMPLICIT OUTPUTS:
0000 176 :
0000 177 : TM_SETUP: COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
0000 178 : ALL PRIVILEGES ACQUIRED.
0000 179 :
0000 180 : COMPLETION CODES:
0000 181 :
0000 182 : EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
0000 183 :
0000 184 : SIDE EFFECTS:
0000 185 :
0000 186 : SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
0000 187 : (VIA RSB) IF ERROR ENCOUNTERED.
0000 188 :
0000 189 :--
0000 190 :
0000 191 :
0000 192 :
0000 193 :

```

```

00000000'EF 00000000'EF 00000000'8F 00000000'EF
03 00 00000000'8F F0 0018 201
0020
0025 202
59 00000000'9F 00 0048 203
00000000'EF 69 DE 004F 204
0056 205
0057 206

TM_SETUP::
CLRL R2 ; INITIALIZE
CLRL R3 ; .. CONDITION
CLRL R4 ; .... TABLE
CLRL R5 ; ..... INDEX
CLRL R6 ; ..... REGISTERS
BSBW MOD MSG PRINT ; PRINT TEST MODULE BEGIN MSG
MOVAL TEST_MOD_SUCC,TMD_ADDR ; ASSUME END MSG WILL SHOW SUCCESS
INSV #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS

MODE TO,58,KRNL ; KERNEL MODE TO ACCESS PHD
MOVL @#CTL$GL,PHD,R9 ; GET PROCESS HEADER ADDRESS
MOVAL PHD$Q PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
MODE FROM,58 ; BACK TO USER MODE
PRIV ADD,ALL ; GET ALL PRIVILEGES

```

Ma
--
-S
-S
-S
TO
89
Th
MA

```

0077 207 $SETPRN S TEST MOD_NAME_D ; SET PROCESS NAME
0084 208 SS_CHECK NORMAL ; CHECK STATUS CODE RETURNED FROM SETPRN
00B2 209 $WAKE S SELFPID ; GET MY PID
00C1 210 SS_CHECK NORMAL ; CHECK FOR NORMAL RETURN
00EF 211 $HIBER S ; UNDO ABOVE WAKE
00F6 212 SS_CHECK NORMAL ; CHECK FOR NORMAL RETURN
0124 213 ;
0124 214 ; THE FOLLOWING CODE ESTABLISHES UIC'S IN THE CONDITION 3 TABLE
0124 215 ;
0124 216 ;
59 00000000'9F n0 0147 217 MODE TO,20$,KRNL ; KERNEL MODE TO ACCESS PCB
59 00BC C9 DO 014E 218 MOVL @#SCH$GL_CURPCB,R9 ; GET CURRENT PCB ADDRESS
0153 219 MOVL PCB$UIC(R9),R9 ; PICK UP UIC FROM PCB
0154 220 MODE FROM,20$ ; ... AND GET BACK TO USER MODE
0154 221 ; R9 NOW CONTAINS 'MY' UIC
0154 222 ;
59 5A 02 9A 0154 223 MOVZBL #2,R10 ; GET COND3 TABLE INDEX NUMBER INTO A REG
00010000 8F C1 0157 224 ADDL3 #^X10000,R9,COND3_E[R10] ; PUT DIFF GROUP UIC INTO 3RD TABLE ELT
0000024F'EF4A 5A D5 0164 225 INCL R10 ; POINT TO 4TH COND3 TABLE ELEMENT
0000024F'EF4A 59 D0 0166 226 MOVL R9,COND3_E[R10] ; PUT MY UIC INTO TABLE
0000024F'EF4A 59 D6 016E 227 INCL R10 ; POINT TO 5TH COND3 TABLE ELEMENT
0000024F'EF4A 59 01 0170 228 ADDL3 #1,R9,COND3_E[R10] ; PUT DIFF MEMBER UIC INTO THE TABLE
0179 229 $CREMBX_S CHAN=MBXCHAN, LOGNAM=SUBJPRN, - ; GET MAILBOX FOR PROCESS
0179 230 MAXMSG=#120, PROMSK=#0, BUFQUO=#240
019E 231 SS_CHECK NORMAL ; CHECK NORMAL COMPLETION
01CC 232 $GETCHN_S CHAN=MBXCHAN, - ; GET CHAN INFO (UNIT NUMBER)
01CC 233 PRIBUF=MBXCHANINFO
01E6 234 SS_CHECK NORMAL ; CHECK NORMAL COMPLETION
0000088'EF 0000020'EF 3C 0214 235 MOVZWL MBXCHANINFO+8+DIB$W_UNIT,MBXUNIT ; SAVE MAILBOX UNIT NUMBER
05 021F 236 RSB ; RETURN TO MAIN ROUTINE
0220 237 TM_CLEANUP:;
0220 238 $DELMBX_S MBXCHAN ; DELETE TERMINATION MAILBOX
FDCF' 30 022E 239 BSBW MOD_MSG_PRINT ; PRINT TEST MODULE END MSG
05 0231 240 RSB ; RETURN TO MAIN ROUTINE

```

```

0232 242 .SBTTL CONDITION SUBROUTINES - SETUP AND CLEANUP
0232 243 :++
0232 244 : FUNCTIONAL DESCRIPTION:
0232 245 :
0232 246 : COND1 AND COND2 CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
0232 247 : BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
0232 248 : CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
0232 249 : ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
0232 250 : CONDITION X TABLE IS INCLUDED IN THE COND1 SUBROUTINE AND CLEANED
0232 251 : UP, IF NECESSARY, IN THE COND2 CLEANUP SUBROUTINE. THIS INCLUDES,
0232 252 : ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
0232 253 : OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
0232 254 : VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
0232 255 : (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
0232 256 :
0232 257 : CALLING SEQUENCE:
0232 258 :
0232 259 : BSBW COND1 BSBW COND2_CLEANUP
0232 260 : WHERE X = 1,2,3,4,5
0232 261 :
0232 262 : INPUT PARAMETERS:
0232 263 :
0232 264 : CONFLICT = 0
0232 265 :
0232 266 : IMPLICIT INPUTS:
0232 267 :
0232 268 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
0232 269 : FOR COND1 TABLES 1,2,3,4,5, RESPECTIVELY.
0232 270 :
0232 271 : OUTPUT PARAMETERS:
0232 272 :
0232 273 : CONFLICT SET TO NON-ZERO IF COND1 TABLE CONFLICT DETECTED.
0232 274 :
0232 275 : IMPLICIT OUTPUTS:
0232 276 :
0232 277 : R2,3,4,5,6 PRESERVED
0232 278 :
0232 279 : COMPLETION CODES:
0232 280 :
0232 281 : NONE
0232 282 :
0232 283 : SIDE EFFECTS:
0232 284 :
0232 285 : NONE
0232 286 :
0232 287 : --
0232 288 :
0232 289 :
0232 290 :
05 0232 291 COND1::
0232 292 RSB ; RETURN TO MAIN ROUTINE
05 0233 293 COND1_CLEANUP::
0233 294 RSB ; RETURN TO MAIN ROUTINE
05 0234 295 COND2::
0234 296 RSB ; RETURN TO MAIN ROUTINE
05 0235 297 COND2_CLEANUP::
0235 298 RSB ; RETURN TO MAIN ROUTINE

```

```

00000170'EF42 0000011C'8F D1 0236 299 COND3::
                20 13 0236 300          CMPL   #SUBJPID,COND1_E[R2] ; NON-ZERO PID SPECIFIED ?
                000001B2'EF43 D5 0242 301          BEQLU  10$ ; YES -- PROCESS IS 'OTHER'
                07 13 0244 302          TSTL  COND2_E[R3] ; IS PROCESS NAME SPECIFIED ?
                02 54 D1 0248 303          BEQL   5$ ; NO -- SUBJECT PROCESS IS 'SELF'
                20 13 024D 304          CMPL  R4,#2 ; DOES CONDITION 3 SPECIFY DIFFERENT GROUP ?
                10 11 0250 305          BEQL  20$ ; YES -- PROCESS NAME FOR DIFF GROUP IS CONF
                0252 306          BRB   10$ ; NO -- MAKE SURE COND 3 SPECIFIES 'OTHER'
                0254 307 5$:
                0254 308 ;
                0254 309 ; PROCESS IS 'SELF'
                0254 310 ;
0000024F'EF44 00000000'EF D1 0254 311          CMPL  ONES,COND3_E[R4] ; DOES CONDITION 3 SPECIFY 'SELF' ?
                1B 13 0260 312          BEQLU  COND3X ; YES -- THEN ALL 3 CONDIT'NS ARE CONSISTENT
                OE 11 0262 313          BRB   20$ ; NO -- INDICATE CONFLICT & GET OUT
                0264 314 10$:
                0264 315 ;
                0264 316 ; PROCESS IS 'OTHER'
                0264 317 ;
0000024F'EF44 00000000'EF D1 0264 318          CMPL  ONES,COND3_E[R4] ; DOES CONDITION 3 SPECIFY 'SELF' ?
                0B 12 0270 319          BNEQU  COND3X ; NO -- THEN ALL 3 CONDITIONS ARE CONSISTENT
                0272 320 20$:
00000000'EF 00000000'EF 90 0272 321          MOVB  ONES,CONFLICT ; YES -- INDICATE CONFLICT
                027D 322 COND3X:
                05 027D 323          RSB ; RETURN TO MAIN ROUTINE
                027E 324 COND3_CLEANUP::
                05 027E 325          RSB ; RETURN TO MAIN ROUTINE
                027F 326 COND4::
                05 027F 327          RSB ; RETURN TO MAIN ROUTINE
                0280 328 COND4_CLEANUP::
                05 0280 329          RSB ; RETURN TO MAIN ROUTINE
                0281 330 COND5::
                05 0281 331          RSB ; RETURN TO MAIN ROUTINE
                0282 332 COND5_CLEANUP::
                05 0282 333          RSB ; RETURN TO MAIN ROUTINE

```

```

0283 335 .SBTTL FORM_CONDS
0283 336 :++
0283 337 : FUNCTIONAL DESCRIPTION:
0283 338 :
0283 339 : FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
0283 340 : THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
0283 341 :
0283 342 : CALLING SEQUENCE:
0283 343 :
0283 344 : BSBW FORM_CONDS
0283 345 :
0283 346 : INPUT PARAMETERS:
0283 347 :
0283 348 : NONE
0283 349 :
0283 350 : IMPLICIT INPUTS:
0283 351 :
0283 352 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
0283 353 : FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
0283 354 : FOR X = 1,2,3,4,5 :
0283 355 : CONDX_T - TITLE TEXT FOR CONDX TABLE
0283 356 : CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
0283 357 : CONDX_C - CONTEXT OF THE CONDX TABLE
0283 358 : CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
0283 359 :
0283 360 : OUTPUT PARAMETERS:
0283 361 :
0283 362 : NONE
0283 363 :
0283 364 : IMPLICIT OUTPUTS:
0283 365 :
0283 366 : NONE
0283 367 :
0283 368 : COMPLETION CODES:
0283 369 :
0283 370 : NONE
0283 371 :
0283 372 : SIDE EFFECTS:
0283 373 :
0283 374 : NONE
0283 375 :
0283 376 :--
0283 377 :
0283 378 :
0283 379 :
0283 380 FORM_CONDS::
0283 381 $FAO_S MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
02A2 382 :
02A2 383 BSBW OUTPUT_MSG : FORMAT CONDITIONS HEADER MSG
14 00 91 02A5 384 CMPB #COND1_C,#NULL : ... AND PRINT IT
03 12 02A8 385 BNEQU 10$ : IS CONDITION 1 NULL ?
00BF 31 02AA 386 BRW FORM_CONDSX : NO -- CONTINUE
: YES -- SUBROUTINE IS FINISHED
02AD 387 10$:
00000000'EF 00000125'EF DE 02AD 388 MOVAL COND1_T,MSG_A : SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF 00000132'EF42 DO 02B8 389 MOVL COND1_TAB[R2],MSG_B : SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
00000000'EF 00 90 02C4 390 MOVB #COND1_C,MSG_CTXT : SAVE CONDITION 1 CONTEXT FOR FAO
02CB 391 MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 : GIVE COND 1 DATA VALUE TO FAO

```

```

14 00 91 02CB 392 BSBW WRITE_MSG2 ; FORMAT AND WRITE CONDITION 1 MSG
03 12 02CE 393 CMPB #COND2_C,#NULL ; IS CONDITION 2 NULL ?
0096 31 02D1 394 BNEQU 20$ ; NO -- CONTINUE
02D3 395 BRW FORM_CONDSX ; YES -- SUBROUTINE IS FINISHED
02D6 396 20$:
00000000'EF 0000017C'EF DE 02D6 397 MOVAL COND2_T,MSG_A ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF 00000192'EF43 DO 02E1 398 MOVL COND2_TAB[R3],MSG_B ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
00000000'EF 00 90 02ED 399 MOVB #COND2_C,MSG_CTXT ; SAVE CONDITION 2 CONTEXT FOR FAO
02F4 400 MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
14 00 30 02F4 401 BSBW WRITE_MSG2 ; FORMAT AND WRITE CONDITION 2 MSG
03 91 02F7 402 CMPB #COND3_C,#NULL ; IS CONDITION 3 NULL ?
006D 12 02FA 403 BNEQU 30$ ; NO -- CONTINUE
31 02FC 404 BRW FORM_CONDSX ; YES -- SUBROUTINE IS FINISHED
02FF 405 30$:
00000000'EF 000001BA'EF DE 02FF 406 MOVAL COND3_T,MSG_A ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF 000001C8'EF44 DO 030A 407 MOVL COND3_TAB[R4],MSG_B ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
00000000'EF 00 90 0316 408 MOVB #COND3_C,MSG_CTXT ; SAVE CONDITION 3 CONTEXT FOR FAO
031D 409 MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
FCE0' 30 031D 410 BSBW WRITE_MSG2 ; FORMAT AND WRITE CONDITION 3 MSG
14 14 91 0320 411 CMPB #COND4_C,#NULL ; IS CONDITION 4 NULL ?
47 13 0323 412 BEQLU FORM_CONDSX ; YES -- SUBROUTINE IS FINISHED
00000000'EF 00000263'EF DE 0325 413 MOVAL COND4_T,MSG_A ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF 00000263'EF45 DO 0330 414 MOVL COND4_TAB[R5],MSG_B ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
00000000'EF 14 90 033C 415 MOVB #COND4_C,MSG_CTXT ; SAVE CONDITION 4 CONTEXT FOR FAO
0343 416 MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
FCBA' 30 0343 417 BSBW WRITE_MSG2 ; FORMAT AND WRITE CONDITION 4 MSG
14 14 91 0346 418 CMPB #COND5_C,#NULL ; IS CONDITION 5 NULL ?
21 13 0349 419 BEQLU FORM_CONDSX ; YES -- SUBROUTINE IS FINISHED
00000000'EF 00000264'EF DE 034B 420 MOVAL COND5_T,MSG_A ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF 00000264'EF46 DO 0356 421 MOVL COND5_TAB[R6],MSG_B ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
00000000'EF 14 90 0362 422 MOVB #COND5_C,MSG_CTXT ; SAVE CONDITION 5 CONTEXT FOR FAO
0369 423 MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
FC94' 30 0369 424 BSBW WRITE_MSG2 ; FORMAT AND WRITE CONDITION 5 MSG
036C 425 FORM_CONDSX:
05 036C 426 RSB ; RETURN TO CALLER

```

```
036D 428 .SBTTL VERIFY
036D 429 :++
036D 430 : FUNCTIONAL DESCRIPTION:
036D 431 :
036D 432 : VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
036D 433 : TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
036D 434 : COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
036D 435 : SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
036D 436 : ($SETPRI). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
036D 437 : BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
036D 438 : AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
036D 439 : COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
036D 440 : ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
036D 441 : THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
036D 442 : PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
036D 443 : WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
036D 444 : AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
036D 445 :
036D 446 : CALLING SEQUENCE:
036D 447 :
036D 448 : BSBW VERIFY
036D 449 :
036D 450 : INPUT PARAMETERS:
036D 451 :
036D 452 : NONE
036D 453 :
036D 454 : IMPLICIT INPUTS:
036D 455 :
036D 456 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
036D 457 : FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
036D 458 : FOR X = 1,2,3,4,5 :
036D 459 : CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
036D 460 : TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
036D 461 : ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
036D 462 : FOR CONDX_E.
036D 463 :
036D 464 : OUTPUT PARAMETERS:
036D 465 :
036D 466 : NONE
036D 467 :
036D 468 : IMPLICIT OUTPUTS:
036D 469 :
036D 470 : VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
036D 471 : IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
036D 472 : ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
036D 473 : AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
036D 474 : ERRORS.
036D 475 :
036D 476 : COMPLETION CODES:
036D 477 :
036D 478 : EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
036D 479 :
036D 480 : SIDE EFFECTS:
036D 481 :
036D 482 : SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
036D 483 : (VIA RSB) IF ERROR ENCOUNTERED.
036D 484 :
```

```

036D 485 :--
036D 486
036D 487
036D 488
036D 489 VERIFY::
00000000'EF 95 036D 490 TSTB CFLAG ; SHOULD CONDITIONS BE PRINTED ?
03 13 0373 491 BEQL 5$ ; NO -- CONTINUE
FF0B 30 0375 492 BSBW FORM_CONDS ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
0378 493 5$:
000011C'EF 0000114'EF D0 0378 494 MOVL SELFPID,SUBJPID ; ASSUME THE SUBJECT PID IS SELF
0000110'EF D4 0383 495 CLRL ZEROPID ; CLEAR ZERO PID
000024F'EF44 00000000'EF D1 0389 496 CMPL ONES,COND3_E[R4] ; IS PROCESS FOR THIS TEST CASE SELF ?
03 12 0395 497 BNEQU 7$ ; NO -- CONTINUE
0074 31 0397 498 BRW 10$ ; YES -- DON'T CREATE A PROCESS
039A 499 7$:
039A 500 $CREPRC_S PIDADR=CREPID, PRCNAM=SUBJPRN, -
039A 501 UIC=COND3_E[R4], IMAGE=IMAGNAM, -
039A 502 MBXUNT=MBXUNIT, QUOTA=QUOTALIST
03D5 503 ; CREATE THE SUBJECT PROCESS
03D5 504 SS CHECK NORMAL ; ... AND MAKE SURE IT CREATED OK
000011C'EF 0000118'EF D0 0403 505 MOVL CREPID,SUBJPID ; MAKE THE SUBJECT PID = THE ONE JUST CREATED
040E 506 10$:
040E 507 $SETPRI_S PIDADR=SUBJPID, PRI=#0, -
040E 508 PRVPRI=ORIGPRI ; GET ORIGINAL PRIORITY
0425 509 SS CHECK NORMAL ; CHECK FOR NORMAL RETURN
0453 510 $SETPRI_S PIDADR=SUBJPID, PRI=ORIGPRI ; RESTORE ORIGINAL PRIORITY
046A 511 SS CHECK NORMAL ; CHECK FOR NORMAL RETURN
000010C'EF 0000170'EF42 D0 0498 512 MOVL COND1_E[R2],DEST_PIDADR ; GET PID ADDRESS OUT OF TABLE
59 00001B2'EF43 D0 04A4 513 MOVL COND2_E[R3],R9 ; PRCNAM ADDR INTO REG FOR INDIRECT REF'RNCE
58 D4 04AC 514 CLRL R8 ; INITIALIZE PRIORITY VALUE REGISTER
04AE 515 15$:
0000110'EF D4 04AE 516 CLRL ZEROPID ; ENSURE ZEROPID IS STILL ZERO
04B4 517 ;
04B4 518 ; ***** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE *****
04B4 519 ;
04B4 520 $SETPRI_S PIDADR=@DEST_PIDADR, PRCNAM=(R9), -
04B4 521 PRI=R8, PRVPRI=ESTPRI
00000000'8F 50 D1 04CB 522 CMPL R0,#SS$_NORMAL ; CODE RECEIVED = CODE EXPECTED ?
61 13 04D2 523 BEQL 18$ ; YES -- CONTINUE
00000000'EF 00000000'8F D0 04D4 524 MOVL #SS$_NORMAL,EXPV ; NO -- LOAD UP EXPECTED AND
00000000'EF 50 D0 04DF 525 MOVL R0,RCV ; ... RECEIVED VALUES, THEN EXIT
04E6 526 ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM SETPRI>
0535 527 18$:
000010C'EF D5 0535 528 TSTL DEST_PIDADR ; PID RETURNED BY SETPRI ?
68 13 053B 529 BEQL 20$ ; NO -- KEEP GOING
000010C'FF 000011C'EF D1 053D 530 CMPL SUBJPID,@DEST_PIDADR ; YES -- IS IT THE CORRECT ONE ?
5B 13 0548 531 BEQL 20$ ; YES -- CONTINUE
00000000'EF 000011C'EF D0 054A 532 MOVL SUBJPID,EXPV ; NO --LOAD UP EXPECTED AND
00000000'EF 000010C'FF D0 0555 533 MOVL @DEST_PIDADR,RCV ; ... RECEIVED VALUES, THEN EXIT
0560 534 ERR_EXIT LONG,<INCORRECT PID RETURNED BY SETPRI>
05A5 535 20$:
58 D5 05A5 536 TSTL R8 ; SETTING PRIORITY 0 ?
09 12 05A7 537 BNEQ 40$ ; NO -- CONTINUE
SA 0000120'EF 90 05A9 538 MOVB ORIGPRI,R10 ; YES -- EXPECT ORIGINAL PRIORITY RETURNED
04 11 05B0 539 BRB 42$ ; .....
58 D5 05B2 540 40$:
SA 58 01 83 05B2 541 SUBB3 #1,R8,R10 ; COMPUTE EXPECTED PRIORITY TO BE RETURNED

```


| | | | | | | | |
|---------------|-------------|----|------|-----|----------|--|---|
| 00000124'EF | 5A | 91 | 05B6 | 542 | 42\$: | | |
| | 5C | 13 | 05B6 | 543 | | CMPB R10,ESTPRI | : IS ESTABLISHED PRIORITY = THAT EXPECTED ? |
| 00000000'EF | 5A | 90 | 05BD | 544 | | BEQLU 50\$ | : YES -- CONTINUE |
| 00000000'EF | 00000124'EF | 90 | 05BF | 545 | | MOVB R10,EXPV | : NO -- LOAD UP EXPECTED AND |
| | | | 05C6 | 546 | | MOVB ESTPRI,RECV | : ... RECEIVED VALUES, THEN EXIT |
| | | | 05D1 | 547 | | ERR_EXIT BYTE,<PRIORITY VALUE NOT | : RETAINED BY SETPRI> |
| FE8D 58 01 1F | | 9D | 061B | 548 | 50\$: | | |
| | | | 061B | 549 | | ACBB #31,#1,R8,15\$ | : LOOP THRU ALL LEGAL PRIORITIES |
| | | | 0621 | 550 | | \$SETPRI_S PIDADR=SUBJPID, PRI=ORIGPRI, - | : - |
| | | | 0621 | 551 | | PRVPRI=ESTPRI | : GET BACK ORIGINAL PRIORITY |
| 00000124'EF | 1F | 91 | 063C | 552 | | SS_CHECK NORMAL | : CHECK FOR NORMAL RETURN |
| | 5C | 13 | 066A | 553 | | CMPB #31,ESTPRI | : DID SETPRI REMEMBER PREVIOUS PRI (31) ? |
| 00000000'EF | 1F | 90 | 0671 | 554 | | BEQLU 60\$ | : YES -- CONTINUE |
| 00000000'EF | 00000124'EF | 90 | 0673 | 555 | | MOVB #31,EXPV | : NO -- LOAD UP EXPECTED AND |
| | | | 067A | 556 | | MOVB ESTPRI,RECV | : ... RECEIVED VALUES, THEN EXIT |
| | | | 0685 | 557 | | ERR_EXIT BYTE,<PRIORITY VALUE NOT | : RETAINED BY SETPRI> |
| 0000011C'EF | 00000118'EF | D1 | 06CF | 558 | 60\$: | | |
| | 03 | 13 | 06CF | 559 | | CMPL CREPID,SUBJPID | : WAS A PROCESS CREATED ? |
| | 0094 | 31 | 06DA | 560 | | BEQLU 65\$ | : YES -- CONTINUE |
| | | | 06DC | 561 | | BRW VERIFYX | : NO -- ALL FINISHED |
| | | | 06DF | 562 | 65\$: | | |
| | | | 06DF | 563 | | \$WAKE_S SUBJPID | : ALLOW CREATED PROCESS TO FINISH |
| | | | 06EE | 564 | | SS_CHECK NORMAL | : CHECK FOR NORMAL STATUS CODE |
| | | | 071C | 565 | | \$QIOW_S CHAN=MBXCHAN, FUNC=#IOS READVBLK, - | : - |
| | | | 071C | 566 | | P1=MBXBUFF+8, P2=MBXBUFF | : - |
| | | | 0745 | 567 | | | : ... AND WAIT FOR IT TO SEND MAIL |
| | | | 0745 | 568 | | SS_CHECK NORMAL | : CHECK FOR NORMAL STATUS CODE |
| | | 05 | 0773 | 569 | VERIFYX: | | |
| | | | 0773 | 570 | | RSB | : RETURN TO CALLER |

```

0774 572 .SBTTL VFY_CLEANUP
0774 573 :++
0774 574 : FUNCTIONAL DESCRIPTION:
0774 575 :
0774 576 : VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
0774 577 : EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
0774 578 : ASSUME THAT VERIFY MAY NOT H'VE EXECUTED IN ITS ENTIRETY (IF AN
0774 579 : ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS CHECK OR ERR_EXIT
0774 580 : ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
0774 581 : IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
0774 582 : WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
0774 583 : POSSIBLY DISCOVERING A SECOND ERROR.
0774 584 :
0774 585 : CALLING SEQUENCE:
0774 586 :
0774 587 : BSBW VFY_CLEANUP
0774 588 :
0774 589 : INPUT PARAMETERS:
0774 590 :
0774 591 : NONE
0774 592 :
0774 593 : IMPLICIT INPUTS:
0774 594 :
0774 595 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
0774 596 : FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
0774 597 : FOR X = 1,2,3,4,5 :
0774 598 : CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
0774 599 : TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
0774 600 : ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
0774 601 : FOR CONDX_E.
0774 602 :
0774 603 : OUTPUT PARAMETERS:
0774 604 :
0774 605 : NONE
0774 606 :
0774 607 : IMPLICIT OUTPUTS:
0774 608 :
0774 609 : NONE
0774 610 :
0774 611 : COMPLETION CODES:
0774 612 :
0774 613 : EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
0774 614 :
0774 615 : SIDE EFFECTS:
0774 616 :
0774 617 : SS CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
0774 618 : (VIA RSB) IF ERROR ENCOUNTERED.
0774 619 :
0774 620 :--
0774 621 :
0774 622 :
0774 623 :
0774 624 VFY_CLEANUP::
0774 625 Cmpl CREPID,SUBJPID ; WAS A PROCESS CREATED FOR THIS TEST CASE ?
0774 626 BNEQU 10$ ; NO -- DON'T DELETE IT
0774 627 $DELPRC_S SUBJPID ; YES -- DELETE IT
0774 628 BRB VFY_CLEANUPX ; ... AND GET OUT

```

```

0000011C'EF 00000118'EF D1
11 12
52 11 0790

```

```

00000120'EF 00000000'EF 91 0792 629 10$:
                                0792 630      CMPB  ONES,ORIGPRI      ; WAS ORIGINAL PRIORITY EVER CHANGED ?
                                13 079D 631      BEQLU  VFY_CLEANUPX    ; NO -- CLEANUP IS COMPLETE
                                079F 632      $SETPRI S PIDADR=SUBJPID, PRI=ORIGPRI ; YES -- GET BACK ORIGINAL
                                07B6 633      SS CHECK NORMAL      ; CHECK FOR NORMAL RETURN
00000120'EF 00000000'EF 90 07E4 634 VFY_CLEANUPX:
                                05 07E4 635      MOVB  ONES,ORIGPRI      ; INDICATE PRI NOT CHANGED YET FOR NEXT T.C.
                                07EF 636      RSB                                ; RETURN TO CALLER
                                07F0 637      .END

```

| | | | | | | | |
|----------------|------------|----|----|-----------------|------------|----|----|
| \$\$\$\$ | = 0000068F | R | 04 | DIB\$W_UNIT | = 0000000C | | |
| \$\$\$CHARS | = 00000025 | | | EFLAG | ***** | X | 04 |
| \$\$\$CHARS1 | = 00000004 | | | ESTPRI | 00000124 | R | 03 |
| \$\$\$CHARS2 | = 0000000A | | | EXPV | ***** | X | 04 |
| \$\$\$CHARS3 | = 00000019 | | | FAO_DESC | ***** | X | 04 |
| \$\$\$CHARS4 | = 00000021 | | | FAO_LEN | ***** | X | 04 |
| \$\$\$CHARS5 | = 00000026 | | | FORM_CONDS | 00000283 | RG | 04 |
| \$\$\$COND_A | = 00000004 | | | FORM_CONDSX | 0000036C | R | 04 |
| \$\$\$STRINGS | = 00000001 | | | IMAGNAM | 00000065 | R | 02 |
| \$\$\$STRINGS2 | = 00000005 | | | IOS_READVBLK | ***** | X | 04 |
| \$\$T1 | = 00000001 | | | LONG | = 00000004 | G | |
| \$\$T2 | = 00000004 | | | MBXBUF | 0000008C | R | 03 |
| BYTE | = 00000001 | G | | MBXCHAN | 00000008 | R | 03 |
| CFLAG | ***** | X | 04 | MBXCHANINFO | 0000000C | R | 03 |
| CHMRTN | ***** | X | 04 | MBXUNIT | 00000088 | R | 03 |
| CHM_CONT | ***** | X | 04 | MOD_MSG_CODE | ***** | X | 04 |
| COMP_SC | ***** | X | 04 | MOD_MSG_PRINT | ***** | X | 04 |
| COND | 00000232 | RG | 04 | MSGT_INP_CTL | 00000019 | R | 02 |
| COND1_C | = 00000000 | | | MSG3_ERR_CTL | 00000039 | RG | 02 |
| COND1_CLEANUP | 00000233 | RG | 04 | MSG_A | ***** | X | 04 |
| COND1_E | 00000170 | R | 03 | MSG_B | ***** | X | 04 |
| COND1_H | 00000131 | RG | 03 | MSG_CTXT | ***** | X | 04 |
| COND1_T | 00000125 | R | 03 | NOTARG | = 00000000 | G | |
| COND1_TAB | 00000132 | R | 03 | NULL | = 00000014 | G | |
| COND2 | 00000234 | RG | 04 | ONES | ***** | X | 04 |
| COND2_C | = 00000000 | | | ORIGPRI | 00000120 | R | 03 |
| COND2_CLEANUP | 00000235 | RG | 04 | OUTPUT_MSG | ***** | X | 04 |
| COND2_E | 00000182 | R | 03 | PCBSL_OIC | = 000000BC | | |
| COND2_H | 00000191 | RG | 03 | PCV | ***** | X | 04 |
| COND2_T | 0000017C | R | 03 | PHD\$Q_PRIVMSK | = 00000000 | | |
| COND2_TAB | 00000192 | R | 03 | PQL\$_BYTLM | = 00000003 | | |
| COND3 | 00000236 | RG | 04 | PQL\$_CPULM | = 00000004 | | |
| COND3X | 0000027D | R | 04 | PQL\$_FILLM | = 00000006 | | |
| COND3_C | = 00000000 | | | PQL\$_LISTEND | = 00000000 | | |
| COND3_CLEANUP | 0000027E | RG | 04 | PQL\$_PGFLQUOTA | = 00000007 | | |
| COND3_E | 0000024F | R | 03 | PQL\$_PRCLM | = 00000008 | | |
| COND3_H | 000001C7 | RG | 03 | PQL\$_TQELM | = 00000009 | | |
| COND3_T | 0000018A | R | 03 | PRIVMASK | = 00000000 | R | 03 |
| COND3_TAB | 000001C8 | R | 03 | PRIV_ARGS | = 00000002 | | |
| COND4 | 0000027F | RG | 04 | PROCESS_ERR | ***** | X | 04 |
| COND4_C | = 00000014 | | | QUAD | = 00000008 | G | |
| COND4_CLEANUP | 00000280 | RG | 04 | QUOTALIST | 00000084 | R | 02 |
| COND4_H | 00000263 | RG | 03 | RECV | ***** | X | 04 |
| COND4_T | 00000263 | R | 03 | REST_REGS | ***** | X | 04 |
| COND4_TAB | 00000263 | R | 03 | SAVE_REGS | ***** | X | 04 |
| COND5 | 00000281 | RG | 04 | SCH\$GL_CURPCB | ***** | X | 04 |
| COND5_C | = 00000014 | | | SELPID | 00000114 | R | 03 |
| COND5_CLEANUP | 00000282 | RG | 04 | SS\$NORMAL | ***** | X | 04 |
| COND5_H | 00000264 | RG | 03 | SUBJPID | 0000011C | R | 03 |
| COND5_T | 00000264 | R | 03 | SUBJPRN | 00000051 | R | 02 |
| COND5_TAB | 00000264 | R | 03 | SUCCESS | ***** | X | 04 |
| CONFLICT | ***** | X | 04 | SYSS\$CMKRN | ***** | GX | 04 |
| CREPID | 00000118 | R | 03 | SYSS\$CREMBX | ***** | GX | 04 |
| CTL\$GL_PHD | ***** | X | 04 | SYSS\$CREPRC | ***** | GX | 04 |
| DESC | = 00000010 | G | | SYSS\$DELMBX | ***** | GX | 04 |
| DEST_PIDADR | 0000010C | R | 03 | SYSS\$DELPRC | ***** | GX | 04 |
| DIB\$R_LENGTH | = 00000074 | | | SYSS\$FAO | ***** | X | 04 |

| | | | |
|-----------------|------------|----|----|
| SYSSGETCHN | ***** | GX | 04 |
| SYSSHIBER | ***** | GX | 04 |
| SYSSQIOW | ***** | GX | 04 |
| SYSSSETPRI | ***** | GX | 04 |
| SYSSSETPRN | ***** | GX | 04 |
| SYSSSETPRV | ***** | GX | 04 |
| SYSSWAKE | ***** | 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 | 00000360 | RG | 04 |
| VERIFYX | 00000773 | R | 04 |
| VFY_CLEANUP | 00000774 | RG | 04 |
| VFY_CLEANUPX | 000007E4 | R | 04 |
| WORD | = 0000C002 | 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 | | | |
| \$ABSS | 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 | 00000265 (613.) | 03 (3.) | NOPIC USR | CON | REL | LCL | NOSHR | NOEXE | RD | WRT | NOVEC | LONG | | | |
| SATSSS45 | 000007F0 (2032.) | 04 (4.) | NOPIC USR | CON | REL | LCL | NOSHR | EXE | RD | WRT | NOVEC | BYTE | | | |

! Performance indicators !

| Phase | Page faults | CPU Time | Elapsed Time |
|------------------------|-------------|-------------|--------------|
| Initialization | 35 | 00:00:00.11 | 00:00:00.33 |
| Command processing | 137 | 00:00:00.66 | 00:00:01.84 |
| Pass 1 | 301 | 00:00:09.60 | 00:00:19.91 |
| Symbol table sort | 0 | 00:00:00.78 | 00:00:01.40 |
| Pass 2 | 139 | 00:00:02.36 | 00:00:05.01 |
| Symbol table output | 16 | 00:00:00.11 | 00:00:00.11 |
| Psect synopsis output | 2 | 00:00:00.03 | 00:00:00.03 |
| Cross-reference output | 0 | 00:00:00.00 | 00:00:00.00 |
| Assembler run totals | 632 | 00:00:13.66 | 00:00:28.63 |

The working set limit was 1500 pages.
 51477 bytes (101 pages) of virtual memory were used to buffer the intermediate code.
 There were 30 pages of symbol table space allocated to hold 490 non-local and 60 local symbols.
 637 source lines were read in Pass 1, producing 26 object records in Pass 2.
 47 pages of virtual memory were used to define 37 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 | 23 |
| TOTALS (all libraries) | 34 |

892 GETS were required to define 34 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:SATSSS45/OBJ=OBJ\$:SATSSS45 MSRC\$:SATSSS45/UPDATE=(ENH\$:SATSSS45)+EXECML\$/LIB+SHRLIB\$:UETP/LIB

0423 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

