


```

SSSSSSSS  AAAAAA  TTTTTTTTTT  SSSSSSSS  SSSSSSSS  SSSSSSSS  44  44  666666
SSSSSSSS  AAAAAA  TTTTTTTTTT  SSSSSSSS  SSSSSSSS  SSSSSSSS  44  44  666666
SS        AA      AA      TT      SS        SS        SS        44  44  66
SS        AA      AA      TT      SS        SS        SS        44  44  66
SS        AA      AA      TT      SS        SS        SS        44  44  66
SS        AA      AA      TT      SS        SS        SS        44  44  66
SSSSSSS   AA      AA      TT      SSSSSS   SSSSSS   SSSSSS   4444444444  66666666
SSSSSSS   AA      AA      TT      SSSSSS   SSSSSS   SSSSSS   4444444444  66666666
          SS  AAAAAAAAAA  TT      SS        SS        SS        44  66  66
          SS  AAAAAAAAAA  TT      SS        SS        SS        44  66  66
          SS  AA      AA      TT      SS        SS        SS        44  66  66
          SS  AA      AA      TT      SS        SS        SS        44  66  66
SSSSSSSS  AA      AA      TT      SSSSSSSS  SSSSSSSS  SSSSSSSS  44  666666  66
SSSSSSSS  AA      AA      TT      SSSSSSSS  SSSSSSSS  SSSSSSSS  44  666666  66

```

```

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)	54	DECLARATIONS
(1)	90	CONDITION TABLES
(1)	106	TM SETUP, TM CLEANUP
(1)	169	CONDITION SUBROUTINES - SETUP AND CLEANUP
(1)	239	FORM CONDS
(1)	332	VERIFY
(1)	469	VFY CLEANUP
(1)	527	CANTIM AST ROUTINE

```

0000 1      .TITLE  SATSSS46 SATS SYSTEM SERVICE TESTS $SETRWM (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 SATSSS46 TO TEST SUCCESSFUL
0000 36 : OPERATION OF THE $SETRWM 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 SSS 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, 1978
0000 47
0000 48 : MODIFIED BY:
0000 49
0000 50 :           : VERSION
0000 51 : 01  -
0000 52 : --

```

```
0000 54 .SBTTL DECLARATIONS
0000 55 :
0000 56 : INCLUDE FILES:
0000 57 :
0000 58 $PRVDEF ; PRIVILEGE BIT DEFINITIONS
0000 59 $PHDDEF ; PROCESS HEADER OFFSETS
0000 60 :
0000 61 : MACROS:
0000 62 :
0000 63 :
0000 64 : EQUATED SYMBOLS:
0000 65 :
00000000 0000 66 ENABLE = 0 ; RESOURCE WAIT MODE ENABLE FLAG
00000001 0000 67 DISABLE = 1 ; RESOURCE WAIT MODE DISABLE FLAG
00000001 0000 68 SPECIAL_TQE = 1 ; REQIDT VALUE FOR $SETIMR SERVICE
00000002 0000 69 LOOP_TQE = 2 ; REQIDT VALUE FOR $SETIMR SERVICE
0000 70 :
0000 71 : OWN STORAGE:
0000 72 :
```

SATSSS46
V04-000

```
00000000 74 .PSECT RODATA, RD, NOWRT, NOEXE, LONG
0000 75 TEST_MOD_NAME:: STRING C, <SATSSS46> ; TEST MODULE NAME
0009 76 TEST_MOD_NAME_D: STRING I, <SATSSS46> ; TEST MODULE NAME DESCRIPTOR
0019 77 MSG1_INP_CTL: STRING I, <SSSRW!4ZW: CONDITIONS:>
0039 78 ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
0039 79 MSG3_ERR_CTL:: STRING I, <*SSSRW!4ZW: !AS>
0051 80 ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
FFFFFFFF DC3CBA00 0051 81 ONE_MIN: .LONG -10*1000*1000*60,-1 ; ONE MINUTE ($SETIMR DELTA)
FFFFFFFF 4D2FA200 0059 82 FIV_MINS: .LONG -10*1000*1000*60*5,-1 ; 5 MINUTES ($SETIMR DELTA)
```

SA
VO

SATSSS46
V04-000

00000000	84		.PSECT	RWDATA,RD,WRT,NOEXE, LONG	
00000008 0000	85	PRIVMASK:	.BLKQ	1	: ADDR OF PRIVILEGE MASK (IN PHD)
0000000A 0008	86	ASTSYNCH:	.BLKW	1	: CONTAINS TESTNUM AFTER AST RTN ENTERED
	87				: ... USED TO VERIFY RES. WAIT REALLY OCCURS
0000000E 000A	88	TQECNT:	.BLKL	1	: CNT OF TIMER REQUESTS (AND, HENCE, TQE'S)

SA
 Sy
 SS
 SS
 SS
 SS
 SS
 AS
 BY
 CA
 CF
 CH
 CH
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CO
 CT
 DE
 DI
 EF
 EN
 EX
 FA
 FA
 FI
 FO
 FO
 LO
 LO
 MO
 MO

```
000E 90 .SBTTL CONDITION TABLES
000E 91 :
000E 92 :
000E 93 :
000E 94 :
000F 95 ***** CONDITION TABLES FOR SETRWM SYSTEM SERVICE *****
000F 96 COND 1,NULL
0010 97 COND 2,NULL
0010 98 COND 3,NULL
0011 99 COND 4,NULL
0011 100 COND 5,NULL
0012 101
0012 102
0013 103
00000000 104 .PSECT SATSSS46, RD, WRT, EXE
```



```

0000 106 .SBTTL TM_SETUP, TM_CLEANUP
0000 107 :++
0000 108 : FUNCTIONAL DESCRIPTION:
0000 109 :
0000 110 : TM SETUP AND TM CLEANUP ARE CALLED TO PERFORM
0000 111 : REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
0000 112 : TEST MODULE EXECUTION.
0000 113 :
0000 114 : CALLING SEQUENCE:
0000 115 :
0000 116 : BSBW TM_SETUP BSBW TM_CLEANUP
0000 117 :
0000 118 : INPUT PARAMETERS:
0000 119 :
0000 120 : NONE
0000 121 :
0000 122 : IMPLICIT INPUTS:
0000 123 :
0000 124 : NONE
0000 125 :
0000 126 : OUTPUT PARAMETERS:
0000 127 :
0000 128 : NONE
0000 129 :
0000 130 : IMPLICIT OUTPUTS:
0000 131 :
0000 132 : TM_SETUP: COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
0000 133 : ALL PRIVILEGES ACQUIRED.
0000 134 :
0000 135 : COMPLETION CODES:
0000 136 :
0000 137 : EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
0000 138 :
0000 139 : SIDE EFFECTS:
0000 140 :
0000 141 : SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
0000 142 : (VIA RSB) IF ERROR ENCOUNTERED.
0000 143 :
0000 144 : --
0000 145 :
0000 146 :
0000 147 :

```

```

0000 148 TM_SETUP::
0000 149 CLRL R2 ; INITIALIZE
0000 150 CLRL R3 ; .. CONDITION
0000 151 CLRL R4 ; .... TABLE
0000 152 CLRL R5 ; ..... INDEX
0000 153 CLRL R6 ; ..... REGISTERS
0000 154 BSBW MOD MSG PRINT ; PRINT TEST MODULE BEGIN MSG
0000 155 MOVAL TEST_MOD_SUCC,TMD_ADDR ; ASSUME END MSG WILL SHOW SUCCESS
0000 156 INSV #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
0000 157 MODE TO,5$,KRNL ; KERNEL MODE TO ACCESS PHD
0000 158 MOVL @#CTL$GL,PHD,R9 ; GET PROCESS HEADER ADDRESS
0000 159 MOVAL PHD$Q,PRIVMSK(R9),PRIVMSK ; GET PRIV MASK ADDRESS
0000 160 MODE FROM,5$ ; BACK TO USER MODE
0000 161 PRIV ADD,ALL ; GET ALL PRIVILEGES

```

```

52 D4 0000
53 D4 0002
54 D4 0004
55 D4 0006
56 D4 0008
FFF3' 30 000A
00000000'EF 00000000'EF DE 000D
03 00 00000000'8F F0 0018
00000000'EF 0020
59 00000000'9F D0 0048
00000000'EF 69 DE 004F
0056 160
0057 161

```

SATSSS46
V04-000

SATS SYSTEM SERVICE TESTS M 11
TM_SETUP, TM_CLEANUP \$SETRWM (SUCC 16-SEP-1984 00:55:58 VAX/VMS Macro V04-00 Page 7
5-SEP-1984 04:31:49 [UETPSY.SRC]SATSSS46.MAR;1 (1)

	0077	162	\$SETPRN S TEST MOD_NAME_D	:	SET PROCESS NAME
	0084	163	SS_CHECK NORMAL	:	CHECK STATUS CODE RETURNED FROM SETPRN
05	0082	164	RSB	:	RETURN TO MAIN ROUTINE
	0083	165	TM_CLEANUP::		
FF4A'	30	0083	BSBW MOD_MSG_PRINT	:	PRINT TEST MODUJE END MSG
	05	0086	RSB	:	RETURN TO MAIN ROUTINE

SA
Tal

```

00B7 169 .SBTTL CONDITION SUBROUTINES - SETUP AND CLEANUP
00B7 170 : **
00B7 171 : FUNCTIONAL DESCRIPTION:
00B7 172 :
00B7 173 : CONDX AND CONDX CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
00B7 174 : BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
00B7 175 : CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
00B7 176 : ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
00B7 177 : CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
00B7 178 : UP, IF NECESSARY, IN THE CONDX CLEANUP SUBROUTINE. THIS INCLUDES,
00B7 179 : ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
00B7 180 : OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
00B7 181 : VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
00B7 182 : (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
00B7 183 :
00B7 184 : CALLING SEQUENCE:
00B7 185 :
00B7 186 : BSBW CONDX BSBW CONDX_CLEANUP
00B7 187 : WHERE X = 1,2,3,4,5
00B7 188 :
00B7 189 : INPUT PARAMETERS:
00B7 190 :
00B7 191 : CONFLICT = 0
00B7 192 :
00B7 193 : IMPLICIT INPUTS:
00B7 194 :
00B7 195 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
00B7 196 : FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
00B7 197 :
00B7 198 : OUTPUT PARAMETERS:
00B7 199 :
00B7 200 : CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
00B7 201 :
00B7 202 : IMPLICIT OUTPUTS:
00B7 203 :
00B7 204 : R2,3,4,5,6 PRESERVED
00B7 205 :
00B7 206 : COMPLETION CODES:
00B7 207 :
00B7 208 : NONE
00B7 209 :
00B7 210 : SIDE EFFECTS:
00B7 211 :
00B7 212 : NONE
00B7 213 :
00B7 214 : --
00B7 215 :
00B7 216 :
00B7 217 :
05 00B7 218 COND1:: ; RETURN TO MAIN ROUTINE
00B7 219 RSB
00B8 220 COND1_CLEANUP:: ; RETURN TO MAIN ROUTINE
05 00B8 221 RSB
00B9 222 COND2:: ; RETURN TO MAIN ROUTINE
05 00B9 223 RSB
00BA 224 COND2_CLEANUP:: ; RETURN TO MAIN ROUTINE
05 00BA 225 RSB

```

```
05 00BB 226 COND3:: RSB ; RETURN TO MAIN ROUTINE
05 00BB 227 RSB ; RETURN TO MAIN ROUTINE
05 00BC 228 COND3_CLEANUP:: RSB ; RETURN TO MAIN ROUTINE
05 00BC 229 RSB ; RETURN TO MAIN ROUTINE
05 00BD 230 COND4:: RSB ; RETURN TO MAIN ROUTINE
05 00BD 231 RSB ; RETURN TO MAIN ROUTINE
05 00BE 232 COND4_CLEANUP:: RSB ; RETURN TO MAIN ROUTINE
05 00BE 233 RSB ; RETURN TO MAIN ROUTINE
05 00BF 234 COND5:: RSB ; RETURN TO MAIN ROUTINE
05 00BF 235 RSB ; RETURN TO MAIN ROUTINE
05 00C0 236 COND5_CLEANUP:: RSB ; RETURN TO MAIN ROUTINE
05 00C0 237 RSB ; RETURN TO MAIN ROUTINE
```

```

OOC1 239 .SBTTL FORM_CONDS
OOC1 240 :++
OOC1 241 : FUNCTIONAL DESCRIPTION:
OOC1 242 :
OOC1 243 : FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
OOC1 244 : THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
OOC1 245 :
OOC1 246 : CALLING SEQUENCE:
OOC1 247 :
OOC1 248 : BSBW FORM_CONDS
OOC1 249 :
OOC1 250 : INPUT PARAMETERS:
OOC1 251 :
OOC1 252 : NONE
OOC1 253 :
OOC1 254 : IMPLICIT INPUTS:
OOC1 255 :
OOC1 256 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
OOC1 257 : FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
OOC1 258 : FOR X = 1,2,3,4,5 :
OOC1 259 : CONDX_T - TITLE TEXT FOR CONDX TABLE
OOC1 260 : CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
OOC1 261 : CONDX_C - CONTEXT OF THE CONDX TABLE
OOC1 262 : CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
OOC1 263 :
OOC1 264 : OUTPUT PARAMETERS:
OOC1 265 :
OOC1 266 : NONE
OOC1 267 :
OOC1 268 : IMPLICIT OUTPUTS:
OOC1 269 :
OOC1 270 : NONE
OOC1 271 :
OOC1 272 : COMPLETION CODES:
OOC1 273 :
OOC1 274 : NONE
OOC1 275 :
OOC1 276 : SIDE EFFECTS:
OOC1 277 :
OOC1 278 : NONE
OOC1 279 :
OOC1 280 :--
OOC1 281 :
OOC1 282 :
OOC1 283 :
OOC1 284 FORM_CONDS::
OOC1 285 $FAO_S MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
OOE0 286 : FORMAT CONDITIONS HEADER MSG
OOE0 287 BSBW OUTPUT_MSG : ... AND PRINT IT
14 FF1D' 30 OOE3 288 CMPB #COND1_C,#NULL : IS CONDITION 1 NULL ?
03 12 OOE6 289 BNEQU 10$ : NO -- CONTINUE
OOC1 290 BRW FORM_CONDSX : YES -- SUBROUTINE IS FINISHED
OOC1 291 10$:
OOE0 292 MOVAL COND1_T,MSG_A : SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
OOE0 293 MOVL COND1_TAB[R2],MSG_B : SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
00000000'EF 0000000E'EF DE OOE0 294 MOVB #COND1_C,MSG_TXT : SAVE CONDITION 1 CONTEXT FOR FAO
00000000'EF 0000000E'EF42 DO OOF6 295 MOV_VAL COND1_C,(COND1_E[R2]),MSG_DATA1 : GIVE COND 1 DATA VALUE TO FAO
00000000'EF 14 90 0102
0109

```

53

21
6E
70
2E

74
4C
41

74
20
64
58

69

73

65
65

```

      FEF4' 30 0109 296      BSBW  WRITE_MSG2      ; FORMAT AND WRITE CONDITION 1 MSG
14  14  91 010C 297      CMPB  #COND2_C,#NULL      ; IS CONDITION 2 NULL ?
      03  12 010F 298      BNEQU  20$              ; NO -- CONTINUE
      0096 31 0111 299      BRW    FORM_CONDSX      ; YES -- SUBROUTINE IS FINISHED
      0114 300 20$:
00000000'EF 0000000F'EF DE 0114 301      MOVAL  COND2_T,MSG_A      ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF 0000000F'EF43 DO 011F 302      MOVL  COND2_TAB[R3],MSG_B ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
      00000000'EF 14 90 012B 303      MOVB  #COND2_C,MSG_CTXT ; SAVE CONDITION 2 CONTEXT FOR FAO
      FE7C' 30 0132 304      MOV VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
14  14  91 0132 305      BSBW  WRITE_MSG2      ; FORMAT AND WRITE CONDITION 2 MSG
      03  12 0135 306      CMPB  #COND3_C,#NULL      ; IS CONDITION 3 NULL ?
      006D 31 0138 307      BNEQU  30$              ; NO -- CONTINUE
      013A 308      BRW    FORM_CONDSX      ; YES -- SUBROUTINE IS FINISHED
      013D 309 30$:
00000000'EF 00000010'EF DE 013D 310      MOVAL  COND3_T,MSG_A      ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF 00000010'EF44 DO 0148 311      MOVL  COND3_TAB[R4],MSG_B ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
      00000000'EF 14 90 0154 312      MOVB  #COND3_C,MSG_CTXT ; SAVE CONDITION 3 CONTEXT FOR FAO
      FEA2' 30 015B 313      MOV VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
14  14  91 015B 314      BSBW  WRITE_MSG2      ; FORMAT AND WRITE CONDITION 3 MSG
      47  13 015E 315      CMPB  #COND4_C,#NULL      ; IS CONDITION 4 NULL ?
00000000'EF 00000011'EF DE 0163 316      BEQLU  FORM_CONDSX      ; YES -- SUBROUTINE IS FINISHED
00000000'EF 00000011'EF45 DO 016E 317      MOVAL  COND4_T,MSG_A      ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
      00000000'EF 14 90 017A 318      MOVL  COND4_TAB[R5],MSG_B ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
      FE7C' 30 0181 319      MOVB  #COND4_C,MSG_CTXT ; SAVE CONDITION 4 CONTEXT FOR FAO
14  14  91 0181 320      MOV VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
      21  13 0184 321      BSBW  WRITE_MSG2      ; FORMAT AND WRITE CONDITION 4 MSG
00000000'EF 00000012'EF DE 0189 322      CMPB  #COND5_C,#NULL      ; IS CONDITION 5 NULL ?
00000000'EF 00000012'EF46 DO 0194 323      BEQLU  FORM_CONDSX      ; YES -- SUBROUTINE IS FINISHED
      00000000'EF 14 90 01A0 324      MOVAL  COND5_T,MSG_A      ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
      FE56' 30 01A7 325      MOVL  COND5_TAB[R6],MSG_B ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
      01AA 326      MOVB  #COND5_C,MSG_CTXT ; SAVE CONDITION 5 CONTEXT FOR FAO
      01A7 327      MOV VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
      01AA 328      BSBW  WRITE_MSG2      ; FORMAT AND WRITE CONDITION 5 MSG
05  01AA 329 FORM_CONDSX: ; RETURN TO CALLER
      01AA 330      RSB

```

```
01AB 332 .SBTTL VERIFY
01AB 333 :++
01AB 334 : FUNCTIONAL DESCRIPTION:
01AB 335 :
01AB 336 : VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
01AB 337 : TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
01AB 338 : COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
01AB 339 : SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
01AB 340 : ($SETRWM). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
01AB 341 : BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
01AB 342 : AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
01AB 343 : COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
01AB 344 : ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
01AB 345 : THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
01AB 346 : PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
01AB 347 : WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
01AB 348 : AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
01AB 349 :
01AB 350 : CALLING SEQUENCE:
01AB 351 :
01AB 352 : BSBW VERIFY
01AB 353 :
01AB 354 : INPUT PARAMETERS:
01AB 355 :
01AB 356 : NONE
01AB 357 :
01AB 358 : IMPLICIT INPUTS:
01AB 359 :
01AB 360 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
01AB 361 : FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
01AB 362 : FOR X = 1,2,3,4,5 :
01AB 363 : CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
01AB 364 : TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
01AB 365 : ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
01AB 366 : FOR CONDX_E.
01AB 367 :
01AB 368 : OUTPUT PARAMETERS:
01AB 369 :
01AB 370 : NONE
01AB 371 :
01AB 372 : IMPLICIT OUTPUTS:
01AB 373 :
01AB 374 : VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
01AB 375 : IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
01AB 376 : ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
01AB 377 : AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
01AB 378 : ERRORS.
01AB 379 :
01AB 380 : COMPLETION CODES:
01AB 381 :
01AB 382 : EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
01AB 383 :
01AB 384 : SIDE EFFECTS:
01AB 385 :
01AB 386 : SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
01AB 387 : (VIA RSB) IF ERROR ENCOUNTERED.
01AB 388 :
```

```

01AB 389 :--
01AB 390
01AB 391
01AB 392
01AB 393
0000000'EF 95 01AB 394 VERIFY:: TSTB CFLAG ; SHOULD CONDITIONS BE PRINTED ?
03 13 01B1 395 BEQL 5$ ; NO -- CONTINUE
FF0B 30 01B3 396 BSBW FORM_CONDS ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
01B6 397 5$:
00000008'EF 0000000'EF B0 01B6 398 MOVW ONES,ASTSYNCH ; INDICATE AST RTN NOT YET EXECUTED
0000000A'EF D4 01C1 399 CLRL TQECNT ; INIT TIMER COUNT FOR THIS TEST CASE
01C7 400 :
01C7 401 : * THE FOLLOWING $SETIMR IS ISSUED TO ALLOCATE (RESERVE) A TIMER
01C7 402 : * QUEUE ENTRY SO THAT IT IS AVAILABLE FOR LATER USE.
01C7 403 :
01C7 404 $SETIMR_S DAYTIM=FIV_MINS, REQIDT=#SPECIAL_TQE
01DA 405 ; ALLOCATE_A TQE BY REQUESTING A TIMER
01DA 406 SS_CHECK NORMAL ; CHECK FOR NORMAL RETURN
0208 407 $SETRWM_S #DISABLE ; DISABLE RESOURCE WAIT MODE
0000000'8F 50 D1 0211 408 CMPL RO,#SS$ _WASCLR ; WAS WAIT MODE PREVIOUSLY ENABLED ?
03 12 0218 409 BNEQU 10$ ; NO -- GO PROCESS ERROR
0063 31 021A 410 BRW TMRLOOP ; YES -- CONTINUE
021D 411 10$:
00000000'EF 00000000'8F D0 021D 412 MOVL #SS$ _WASCLR,EXPV ; LOAD UP EXPECTED AND ...
00000000'EF 50 D0 0228 413 MOVL RO,RECV ; ... RECEIVED VALUES, THEN EXIT
022F 414 ERR_EXIT LONG,<RESOURCE WAIT MODE WAS NOT INITIALLY ENABLED>
0280 415 :
0280 416 : * THE FOLLOWING LOOP USES TIMER QUEUE ENTRIES UNTIL QUOTA
0280 417 : * IS EXHAUSTED, AT WHICH TIME $SETIMR WILL RETURN EXQUOTA.
0280 418 :
0280 419 TMRLOOP:
0000000A'EF D6 0280 420 INCL TQECNT ; INCREMENT COUNT OF TIMER REQUESTS
0286 421 $SETIMR_S DAYTIM=FIV_MINS, REQIDT=#LOOP_TQE
0299 422 ; ENTER_A TIMER REQUEST
00000000'8F 50 D1 0299 423 CMPL RO,#SS$ _NORMAL ; TIMER REQUEST ACCEPTED ?
DE 13 02A0 424 BEQLU TMRLOOP ; YES -- GO DO ANOTHER
02A2 425 SS_CHECK EXQUOTA ; NO -- TERMINATE TEST MODULE IF NOT EXQUOTA
02D0 426 :
02D0 427 : * AT THIS POINT THE TIMER QUEUE ENTRY QUOTA SHOULD BE EXHAUSTED.
02D0 428 : * NOW, WE WILL ENABLE RESOURCE WAIT MODE AND RE-ISSUE THE $SETIMR
02D0 429 : * WHICH FAILED ABOVE. THIS TIME, A RESOURCE WAIT WILL ENSUE; IT
02D0 430 : * WILL BE RESOLVED IN AN AST ROUTINE BY CANCELING ALL TIMER REQUESTS.
02D0 431 :
02D0 432 $CANTIM S REQIDT=#SPECIAL_TQE ; FREE UP SPECIAL RESERVED TQE FOR RE-USE BE
02DB 433 SS_CHECK NORMAL ; CHECK FOR NORMAL RETURN
0309 434 $SETIMR_S DAYTIM=ONE_MIN, ASTADR=CANTIM_AST, -
0309 435 REQIDT=#SPECIAL_TQE ; SCHEDULE AST TO FREE RESOURCE WAIT
0320 436 SS_CHECK NORMAL ; CHECK FOR NORMAL RETURN
034E 437 :
034E 438 : ***** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE *****
034E 439 :
00000000'8F 50 D1 034E 440 $SETRWM_S #ENABLE ; ENABLE RESOURCE WAIT MODE
03 12 0357 441 CMPL RO,#SS$ _WASSET ; WAS WAIT MODE PREVIOUSLY DISABLED ?
0061 31 035E 442 BNEQU 20$ ; NO -- IT SHOULD HAVE BEEN
0363 443 BRW 30$ ; YES -- CONTINUE
00000000'EF 00000000'8F D0 0363 444 20$:
MOVL #SS$ _WASSET,EXPV ; LOAD UP EXPECTED AND ...

```



```

00000000'EF 50 D0 036E 446      MOVL  R0,RECV      ; ... RECEIVED VALUES, THEN EXIT
                   0375 447      ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM SETRWM>
                   03C4 448 30$:
                   03C4 449      ;
                   03C4 450      ; * THE FOLLOWING CODE ISSUES ONE MORE SETIMR, WHICH SHOULD
                   03C4 451      ; * HAVE TO WAIT FOR RESOURCES, SINCE WE HAVE JUST GOBBLED THEM
                   03C4 452      ; * UP. TO ENSURE THAT THE SETIMR DOES NOT WAIT FOREVER, AN
                   03C4 453      ; * AST ROUTINE WAS SCHEDULED (ABOVE) TO BE DELIVERED IN 1 MINUTE;
                   03C4 454      ; * IT WILL CANCEL ALL ACTIVE TIMER REQUESTS; THIS SHOULD
                   03C4 455      ; * FREE THE RESOURCES NEEDED BY OUR SETIMR, WHICH SHOULD,
                   03C4 456      ; * IN TURN, COMPLETE NORMALLY.
                   03C4 457      ;
                   03C4 458      $SETIMR_S DAYTIM=FIV_MINS, REQIDT=#LOOP TOE
                   03D7 459      ; TRY TO REQUEST A TIMER
00000008'EF 00000000'EF B1 03D7 460      CMPW  TESTNUM,ASTSYNCH ; WAS AST ROUTINE ENTERED ?
                   13 03E2 461      BEQLU 40$ ; YES -- GO CHECK RETURN FROM SETIMR
00000000'EF 00000000'EF B0 03E4 462      MOVW  TESTNUM,EXPV ; NO -- LOAD UP EXPECTED AND ...
00000000'EF 00000008'EF B0 03EF 463      MOVW  ASTSYNCH,RECV ; ... RECEIVED VALUES, THEN EXIT
                   03FA 464      ERR_EXIT WORD,<RESOURCE WAIT DID NOT OCCUR AS EXPECTED>
                   0446 465 40$:
                   0446 466      SS CHECK NORMAL ; SETIMR SHOULD EVENTUALLY FINISH NORMALLY
05 0474 467      RSB ; RETURN TO CALLER

```

```

0475 469 .SBTTL VFY_CLEANUP
0475 470 :++
0475 471 : FUNCTIONAL DESCRIPTION:
0475 472 :
0475 473 : VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
0475 474 : EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
0475 475 : ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
0475 476 : ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS CHECK OR ERR_EXIT
0475 477 : ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
0475 478 : IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
0475 479 : WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
0475 480 : POSSIBLY DISCOVERING A SECOND ERROR.
0475 481 :
0475 482 : CALLING SEQUENCE:
0475 483 :
0475 484 : BSBW VFY_CLEANUP
0475 485 :
0475 486 : INPUT PARAMETERS:
0475 487 :
0475 488 : NONE
0475 489 :
0475 490 : IMPLICIT INPUTS:
0475 491 :
0475 492 : R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
0475 493 : FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
0475 494 : FOR X = 1,2,3,4,5 :
0475 495 : COND_X E - ADDRESS OF TABLE OF DATA VALUES FOR COND_X
0475 496 : TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
0475 497 : ARGUMENT, THE ARGUMENT1 NAME MAY BE USED AS A SYNONYM
0475 498 : FOR COND_X_E.
0475 499 :
0475 500 : OUTPUT PARAMETERS:
0475 501 :
0475 502 : NONE
0475 503 :
0475 504 : IMPLICIT OUTPUTS:
0475 505 :
0475 506 : NONE
0475 507 :
0475 508 : COMPLETION CODES:
0475 509 :
0475 510 : EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
0475 511 :
0475 512 : SIDE EFFECTS:
0475 513 :
0475 514 : SS CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
0475 515 : (VIA RSB) IF ERROR ENCOUNTERED.
0475 516 :
0475 517 :--
0475 518 :
0475 519 :
0475 520 :
0475 521 VFY_CLEANUP::
0475 522 $CANTIM_S REQIDT=#SPECIAL TQE ; CANCEL AST TIMER IF STILL PENDING
0480 523 $CANTIM_S REQIDT=#LOOP_TQE ; MAKE SURE ALL OTHER TIMER REQ'STS ARE GONE
0488 524 SS CHECK NORMAL ; ... AND CHECK FOR NORMAL COMPLETION
05 0489 525 RSB ; RETURN TO CALLER

```

```

04BA 527 .SBTTL CANTIM AST ROUTINE
04BA 528 :
04BA 529 : THE AST ROUTINE IS SCHEDULED AFTER IT IS DISCOVERED
04BA 530 : THROUGH A FAILING $SETIMR THAT THE TIMER QUEUE
04BA 531 : ENTRY QUOTA HAS BEEN EXHAUSTED BY REPEATED
04BA 532 : SETIMR'S. THEN, WITH RESOURCE WAIT MODE
04BA 533 : ENABLED, THE SETIMR IS REPEATED; A RESOURCE WAIT
04BA 534 : IS EXPECTED. THIS AST ROUTINE SHOULD BE DELIVERED
04BA 535 : DURING THE RESOURCE WAIT; IT WILL CANCEL ALL ACTIVE
04BA 536 : TIMER REQUESTS IN ORDER TO CLEAR THE RESOURCE WAIT
04BA 537 : CONDITION. THE "SUSPENDED" SETIMR SHOULD THEN FINISH
04BA 538 : NORMALLY. THE ASTSYNCH DATA BASE IS SET TO TESTNUM
04BA 539 : IN THIS ROUTINE TO INDICATE THAT DELIVERY HAS OCCURRED.
04BA 540 : THIS IS VERIFIED IN THE MAIN ROUTINE TO GUARANTEE
04BA 541 : THAT A WAIT DID INDEED OCCUR -- I.E., THE SETIMR DID
04BA 542 : NOT COMPLETE IMMEDIATELY, BUT INSTEAD WAITED ONE
04BA 543 : MINUTE UNTIL THE AST WAS DELIVERED.
04BA 544 :

```

```

00000008'EF 00000000'EF 0000 04BA 545 CANTIM_AST: .WORD 0
04BC 546 MOVW TESTNUM,ASTSYNCH ; INDICATE AST RTN EXECUTED FOR THIS T.C.
04C7 547 $CANTIM_S REQIDT=#LOOP_TQE ; CANCEL ALL OUTSTANDING TIMER REQUESTS
04D2 548 RET ; RETURN TO MAIN ROUTINE
04D3 549 .END

```

SSSS	= 00000404	R	04	MSG1_INP_CTL	00000019	R	02
SSSCHARS	= 00000027			MSG3_ERR_CTL	00000039	RG	02
SSSTRINGS	= 00000001			MSG_A	*****	X	04
SS1	= 00000000			MSG_B	*****	X	04
SS2	= 00000004			MSG_CTXT	*****	X	04
ASTSYNCH	00000008	R	03	NOTARG	= 00000000	G	
BYTE	= 00000001	G		NULL	= 00000014	G	
CANTIM_AST	000004BA	R	04	ONES	*****	X	04
CFLAG	*****	X	04	ONE_MIN	00000051	R	02
CHMRTN	*****	X	04	OUTPUT_MSG	*****	X	04
CHM_CONT	*****	X	04	PCV	*****	X	04
COMP_SC	*****	X	04	PHDSQ_PRIVMSK	= 00000000		
CONDT	000000B7	RG	04	PRIVMSK	00000000	R	03
COND1_C	= 00000014			PRIV_ARGS	= 00000002		
COND1_CLEANUP	000000B8	RG	04	PROCESS_ERR	*****	X	04
COND1_H	0000000E	RG	03	QUAD	= 00000008	G	
COND1_T	0000000E	R	03	RECV	*****	X	04
COND1_TAB	0000000E	R	03	REST_REGS	*****	X	04
COND2	000000B9	RG	04	SAVE_REGS	*****	X	04
COND2_C	= 00000014			SPECIAL_TQE	= 00000001		
COND2_CLEANUP	000000BA	RG	04	SS\$_EXQOTA	*****	X	04
COND2_H	0000000F	RG	03	SS\$_NORMAL	*****	X	04
COND2_T	0000000F	R	03	SS\$_WASCLR	*****	X	04
COND2_TAB	0000000F	R	03	SS\$_WASSET	*****	X	04
COND3	000000BB	RG	04	SUCCESS	*****	X	04
COND3_C	= 00000014			SYSSCANTIM	*****	GX	04
COND3_CLEANUP	000000BC	RG	04	SYSSCMKRN	*****	GX	04
COND3_H	00000010	RG	03	SYSSFAO	*****	X	04
COND3_T	00000010	R	03	SYSSSETIMR	*****	GX	04
COND3_TAB	00000010	R	03	SYSSSETPRN	*****	GX	04
COND4	000000BD	RG	04	SYSSSETPRV	*****	GX	04
COND4_C	= 00000014			SYSSSETRWM	*****	GX	04
COND4_CLEANUP	000000BE	RG	04	TESTNUM	*****	X	04
COND4_H	00000011	RG	03	TEST_MOD_NAME	00000000	RG	02
COND4_T	00000011	R	03	TEST_MOD_NAME_D	00000009	R	02
COND4_TAB	00000011	R	03	TEST_MOD_SUCC	*****	X	04
COND5	000000BF	RG	04	YMD_ADDR	*****	X	04
COND5_C	= 00000014			TMR[OOP	00000280	R	04
COND5_CLEANUP	000000C0	RG	04	TM_CLEANUP	000000B3	RG	04
COND5_H	00000012	RG	03	TM_SETUP	00000000	RG	04
COND5_T	00000012	R	03	TQECNT	0000000A	R	03
COND5_TAB	00000012	R	03	VERIFY	000001AB	RG	04
CTL\$GC_PHD	*****	X	04	VFY_CLEANUP	00000475	RG	04
DESC	= 00000010	G		WORD	= 00000002	G	
DISABLE	= 00000001			WRITE_MSG2	*****	X	04
EFLAG	*****	X	04				
ENABLE	= 00000000						
EXPV	*****	X	04				
FAO_DESC	*****	X	04				
FAO_LEN	*****	X	04				
FIV_MINS	00000059	R	02				
FORM_CONDS	000000C1	RG	04				
FORM_CONDSX	000001AA	R	04				
LONG	= 00000004	G					
LOOP_TQE	= 00000002						
MOD_MSG_CODE	*****	X	04				
MOD_MSG_PRINT	*****	X	04				

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABSS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
RODATA	00C00061 (97.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	00000013 (19.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS46	000004D3 (1235.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.05	00:00:00.33
Command processing	107	00:00:00.68	00:00:02.13
Pass 1	236	00:00:05.63	00:00:11.60
Symbol table sort	0	00:00:00.44	00:00:00.53
Pass 2	116	00:00:01.56	00:00:02.16
Symbol table output	12	00:00:00.08	00:00:00.11
Psect synopsis output	3	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	505	00:00:08.47	00:00:16.89

The working set limit was 1500 pages.
29546 bytes (58 pages) of virtual memory were used to buffer the intermediate code.
There were 20 pages of symbol table space allocated to hold 296 non-local and 27 local symbols.
549 source lines were read in Pass 1, producing 23 object records in Pass 2.
31 pages of virtual memory were used to define 26 macros.

! Macro library statistics !

Macro library name	Macros defined
-\$255\$DUA28:[SHRLIB]UETP.MLB;1	7
-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	1
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2	15
TOTALS (all libraries)	23

526 GETS were required to define 23 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:SATSSS46/OBJ=OBJ\$:SATSSS46 MSRC\$:SATSSS46/UPDATE=(ENH\$:SATSSS46)+EXECMLS/LIB+SHRLIB\$:UETP/LIB

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

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

