

(2)	52	DECLARATIONS
(2)	56	MACROS
(3)	370	DATA STORAGE AND MESSAGE STRINGS
(6)	582	INITIALIZATION
(7)	685	FORCE ERRORS IN CRETVA
(8)	708	FORCE ERRORS FROM EXPREG
(9)	724	SUBROUTINES TO CALL THE SERVICES
(10)	917	MISCELLANEOUS SUBROUTINES

```
0000 1 :  
0000 2 : MEMORY MANAGEMENT SERVICES TEST #7  
0000 3 :  
0000 4 :  
0000 5 : .TITLE MMGXQUOTA - TEST MMG SERVICES WITH LOW PGFLQUOTA  
0000 6 : .IDENT 'V04-000'  
0000 7 :  
0000 8 : *****  
0000 9 : *  
0000 10 : * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *  
0000 11 : * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *  
0000 12 : * ALL RIGHTS RESERVED. *  
0000 13 : *  
0000 14 : * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *  
0000 15 : * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *  
0000 16 : * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *  
0000 17 : * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *  
0000 18 : * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *  
0000 19 : * TRANSFERRED. *  
0000 20 : *  
0000 21 : * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *  
0000 22 : * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *  
0000 23 : * CORPORATION. *  
0000 24 : *  
0000 25 : * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *  
0000 26 : * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *  
0000 27 : *  
0000 28 : *  
0000 29 : *****  
0000 30 :  
0000 31 : ++  
0000 32 : FACILITY: USER MODE MEMORY MANAGEMENT SERVICES TEST  
0000 33 :  
0000 34 : ABSTRACT: THIS SET OF ROUTINES TESTS THE MEMORY MANAGEMENT SERVICES  
0000 35 :  
0000 36 : ENVIRONMENT: USER MODE DIAGNOSTIC  
0000 37 :  
0000 38 : AUTHOR: PETER H. LIPMAN , CREATION DATE: 6-JAN-77  
0000 39 :  
0000 40 : MODIFIED BY:  
0000 41 :  
0000 42 : V02-012 SHZ0008 Stephen Zalewski 20-Aug-1980  
0000 43 : Added further tests to system services tested in this  
0000 44 : program. Also incorporated program into MMG test  
0000 45 : package.  
0000 46 :  
0000 47 : V02-012 TSC0007 Tom Clark 25-Jul-1980  
0000 48 : Added further tests to system services tested in this  
0000 49 : program.  
0000 50 :
```

```

0000 52      .SBTTL  DECLARATIONS
0000 53      :
0000 54      : INCLUDE FILES:
0000 55      :
0000 56      .SBTTL  MACROS
0000 57      :
0000 58      : MACROS:
0000 59      :
0000 60      .MACRO  LIST
0000 61      .LIST  MEB
0000 62      .ENDM  LIST
0000 63
0000 64      .MACRO  NLIST
0000 65      .NLIST  MEB
0000 66      .ENDM  NLIST
0000 67
0000 68      .MACRO  READ  SIZ=#1,ADR=(R2),?L1,?L2
0000 69      IFNORD  <SIZ>,<ADR>,L1
0000 70      BRB      L2
0000 71 L1:    MOVAL  W^READERR,R1
0000 72      BSBW    PROBERR
0000 73 L2:
0000 74      .ENDM  READ
0000 75
0000 76      .MACRO  WRITE SIZ=#1,ADR=(R2),?L1,?L2
0000 77      IFNOWRT <SIZ>,<ADR>,L1
0000 78      BRB      L2
0000 79 L1:    MOVAL  W^WRITERR,R1
0000 80      BSBW    PROBERR
0000 81 L2:
0000 82      .ENDM  WRITE
0000 83
0000 84      .MACRO  NOREAD SIZ=#1,ADR=(R2),?L1
0000 85      IFNORD  <SIZ>,<ADR>,L1
0000 86      MOVAL  W^NOREADERR,R1
0000 87      BSBW    PROBERR
0000 88 L1:
0000 89      .ENDM  NOREAD
0000 90
0000 91      .MACRO  NOWRITE SIZ=#1,ADR=(R2),?L1
0000 92      IFNOWRT <SIZ>,<ADR>,L1
0000 93      MOVAL  W^NOWRITERR,R1
0000 94      BSBW    PROBERR
0000 95 L1:
0000 96      .ENDM  NOWRITE
0000 97
0000 98      .MACRO  ADJWSL  PAGCNT,LIMIT=#0,WSETLM,STATUS=S^#SSS_NORMAL
0000 99      LIST
0000 100     MOVZWL  STATUS,R3
0000 101     MOVL    PAGCNT,R4
0000 102     MOVL    LIMIT,R5
0000 103     .IF     B,WSETLM
0000 104     .IF     DIF,<LIMIT>,<#0>
0000 105     MOVAL  W^WRKSETLIM,R6
0000 106     .IFF
0000 107     CLRL    R6
0000 108     .ENDC

```

```

0000 109      .IFF
0000 110      MOVAL  WSETLM,R6
0000 111      .ENDC
0000 112      BSBW   ADJWSLSUBR
0000 113      NLIST
0000 114      .ENDM  ADJWSL
0000 115
0000 116      .MACRO EXPREG  PAGCNT,REGION=#0,STATUS=S^#SS$_NORMAL,-
0000 117      LIST    RETADR=W^RETRANGE
0000 118
0000 119      MOVZWL STATUS,R3
0000 120      MOVL   PAGCNT,R4
0000 121      MOVAL  RETADR,R1
0000 122      .IF    IDN,<REGION>,<#0>
0000 123      CLRL   R5
0000 124      .IFF
0000 125      MOVL   REGION,R5
0000 126      .ENDC
0000 127      BSBW   EXPREGSUBR
0000 128      NLIST
0000 129      .ENDM  EXPREG
0000 130
0000 131      .MACRO CNTREG  PAGCNT,REGION=#0,STATUS=S^#SS$_NORMAL,-
0000 132      LIST    RETADR=W^RETRANGE
0000 133
0000 134      MOVZWL STATUS,R3
0000 135      MOVL   PAGCNT,R4
0000 136      MOVAL  RETADR,R1
0000 137      .IF    IDN,<REGION>,<#0>
0000 138      CLRL   R5
0000 139      .IFF
0000 140      MOVL   REGION,R5
0000 141      .ENDC
0000 142      BSBW   CNTREGSUBR
0000 143      NLIST
0000 144      .ENDM  CNTREG
0000 145
0000 146      .MACRO LKWSET STARTVA,ENDVA,STATUS=S^#SS$_WASCLR,-
0000 147      LIST    INADR=W^INRANGE,RETADR=W^RETRANGE
0000 148
0000 149      .IF    NB,STARTVA
0000 150      MOVL  STARTVA,W^INRANGE
0000 151      .ENDC
0000 152      .IF    NB,ENDVA
0000 153      MOVL  ENDVA,W^INRANGE+4
0000 154      .ENDC
0000 155      MOVZWL STATUS,R3
0000 156      MOVAL  INADR,R0
0000 157      MOVAL  RETADR,R1
0000 158      BSBW   LKWSETSUBR
0000 159      NLIST
0000 160      .ENDM  LKWSET
0000 161
0000 162      .MACRO ULWSET STARTVA,ENDVA,STATUS=S^#SS$_WASET,-
0000 163      LIST    INADR=W^INRANGE,RETADR=W^RETRANGE
0000 164
0000 165      .IF    NB,STARTVA

```

```
0000 166          MOVL  STARTVA,W^INRANGE
0000 167          .ENDC
0000 168          .IF   NB,ENDVA
0000 169          MOVL  ENDVA,W^INRANGE+4
0000 170          .ENDC
0000 171          MOVZWL STATUS,R3
0000 172          MOVAL  INADR,R0
0000 173          MOVAL  RETADR,R1
0000 174          BSBW   ULWSETSUBR
0000 175          NLIST
0000 176          .ENDM  ULWSET
0000 177
0000 178          .MACRO CRETVA STARTVA,ENDVA,STATUS=S^#SS$ NORMAL,-
0000 179                  INADR=W^INRANGE,RETADR=W^RETRANGE
0000 180          LIST
0000 181          .IF   NB,STARTVA
0000 182          MOVL  STARTVA,W^INRANGE
0000 183          .ENDC
0000 184          .IF   NB,ENDVA
0000 185          MOVL  ENDVA,W^INRANGE+4
0000 186          .ENDC
0000 187          MOVZWL STATUS,R3
0000 188          MOVAL  INADR,R0
0000 189          MOVAL  RETADR,R1
0000 190          BSBW   CRETVASUBR
0000 191          NLIST
0000 192          .ENDM  CRETVA
0000 193
0000 194          :*****
0000 195          :      THIS MACRO DELETES THE CURRENT VA ADDRESSES.  ALSO HIDDEN IN THIS
0000 196          :      MACRO IS A TEST TO SEE IF A DELETE GLOBAL SECTION MUST ALSO BE
0000 197          :      INVOKED.  THIS OCCURS DURING THE SECOND AND THIRD RUNS OF THE TEST
0000 198          :      PROGRAM WHEN GLOBAL SECTION MAPPING AND PFN MAPPING ARE USED.
0000 199          :*****
0000 200          :
0000 201          .MACRO DELTVA STARTVA,ENDVA,STATUS=S^#SS$ NORMAL,-
0000 202                  INADR=W^INRANGE,RETADR=W^RETRANGE,?L1
0000 203          LIST
0000 204          CMPW   W^WHICHRUN,#RUN1          ;IF USING MAPPED SECTIONS FOR TESTS,
0000 205          BLEQ  L1                          ;DELETE SECTION AND THEN DO A DELTVA
0000 206          $DGBLSC_S -
0000 207          GSDNAM=<W^GBLSECNAM>
0000 208          L1:
0000 209          .IF   NB,STARTVA
0000 210          MOVL  STARTVA,W^INRANGE
0000 211          .ENDC
0000 212          .IF   NB,ENDVA
0000 213          MOVL  ENDVA,W^INRANGE+4
0000 214          .ENDC
0000 215          MOVZWL STATUS,R3
0000 216          MOVAL  INADR,R0
0000 217          MOVAL  RETADR,R1
0000 218          BSBW   DELTVASUBR
0000 219
0000 220          NLIST
0000 221          .ENDM  DELTVA
0000 222
```

```

0000 223 :*****
0000 224 : * 3) MACRO USED TO CREATE PERMANENT GLOBAL SECTIONS. THE FILE NAME
0000 225 : AND GBL SECTION DEVICE NAME ARE DECLARED INDEPENDANTLY OF THIS MACRO
0000 226 : AND ARE PUSHED ONTO THE STACK MANUALLY BEFORE THE $CRMPSC SERVICE IS
0000 227 : CALLED.
0000 228 : ALSO NOTE THE TEMPORARY FIX LOCATED AT THE BOTTOM OF THE MACRO. THE
0000 229 : FIX IS NEEDED BECAUSE THE LAST GLOBAL SECTION MAY NOT HAVE BEEN DELETED
0000 230 : BY THE TIME THE NEXT MAPPED SECTION IS CREATED. IN THIS CASE THE
0000 231 : RETURNED ERROR IS TRAPPED AND THE SECTION IS REMAPPED UNTIL THE
0000 232 : SUCCESS CODE IS RETURNED. THIS BUG IS SLATED TO BE FIXED. WHEN THAT
0000 233 : IS ACCOMPLISHED REFER TO THE COMMENTS ABOVE THE FIX TO DETERMINE WHAT
0000 234 : LINES SHOULD BE REMOVED TO DELETE THE PATCH.
0000 235 :*****
0000 236 :
0000 237 : .MACRO CREMAPSEC -
0000 238 : STARTVA ;-
0000 239 : ENDVA ;-
0000 240 : STATUS ;-
0000 241 : INADR=<W^INRANGE>,- ; INPUT RANGE
0000 242 : RETADR=<W^RETRANGE> ,- ; RETURN RANGE
0000 243 : FLAGS=#0 ; SECTION FLAGS
0000 244 : GSDNAM=<W^GBLSECNAM>,- ; GLOBAL SECTION NAME
0000 245 : RELPAG=#0 ; RELATIVE PAGE
0000 246 : FILNAM=<W^FILENAME>,- ; FILE NAME
0000 247 : PAGCNT=#0 ; MAX SIZE OF SECTION
0000 248 : VBN=#0 ;- ; STARTING WIRT BLK NO. IN FILE
0000 249 : ?L1
0000 250 : LIST
0000 251 : .IF NB,STARTVA
0000 252 : MOVL STARTVA,W^INRANGE
0000 253 : .ENDC
0000 254 : .IF NB,ENDVA
0000 255 : MOVL ENDVA,W^INRANGE+4
0000 256 : .ENDC
0000 257 : MOVZWL STATUS,R3
0000 258 L1:
0000 259 : PUSHL VBN
0000 260 : PUSHL PAGCNT
0000 261 : $PUSHADR FILNAM
0000 262 : PUSHL RELPAG
0000 263 : $PUSHADR GSDNAM
0000 264 : PUSHL FLAGS
0000 265 : $PUSHADR RETADR
0000 266 : $PUSHADR INADR
0000 267 : CALLS #8,LIB$_CREMAPSEC
0000 268 :*****
0000 269 : IN ORDER TO DELTE THE PATCH MFNTIONED ABOVE SIMPLY REMOVE THE LINES FROM
0000 270 : HERE TO THE NEXT SET OF ASTERICS.
0000 271 :
0000 272 : CMPL #^X1828A,R0 ; IF PREVIOUS GBL SECTION HASN'T
0000 273 : BEQL L1 ; BEEN DELETED TRY CREMAPSEC AGAIN.
0000 274 :*****
0000 275 : MOVAL W^CRMPSCERR,R1
0000 276 : BSBW CHECK1
0000 277 : NLIST
0000 278 : .ENDM CREMAPSEC
0000 279

```



```

0000 280 :*****
0000 281 :      THIS MACRO IS USED TO DETERMINE WHETHER THE MAKEVA SHOULD BE
0000 282 :      TRANSLATED INTO A $CRETVA, A $CRMPSC USING GLOBAL SECTIONS,
0000 283 :      OR A $CRMPSC USING PFN MAPPING.  THE DECISION IS BASED UPON WHICH
0000 284 :      RUN THE PROGRAM IS CURRENTLY IN.
0000 285 :      ALSO NOTE THAT THE FIX THAT WAS APPLIED TO $CRMPSC MACRO LOCATED
0000 286 :      ABOVE, WAS ALSO APPLIED TO THE $CRMPSC CALL LOCATED IN THE THIRD
0000 287 :      PART OF THIS MACRO.  UPON THE FIX OF THIS BUG REFER TO THE
0000 288 :      FIXED AREA TO FIND OUT WHAT MUST BE DELETED IN ORDER TO GET RID
0000 289 :      OF THE TEMPORARY PATCH.
0000 290 :*****
0000 291 :
0000 292 :      .MACRO MAKEVA  STARTVA,ENDVA,STAT=S^#SS$_NORMAL,?L1,?L2,?L10,?L20,-
0000 293 :                   ?L30,?L40
0000 294 :      LIST
0000 295 :          CASEL W^WHICHRUN,#1,#RUNMAX
0000 296 :      L1:
0000 297 :          .WORD L10-L1                ;IF FIRST RUN USE CRETVA.
0000 298 :          .WORD L20-L1                ;IF SECOND USE GBL CREMAPSEC
0000 299 :          .WORD L30-L1                ;IF THIRD USE PFN MAPPING
0000 300 :      L10:
0000 301 :          CRETVA STARTVA,ENDVA,STATUS=STAT
0000 302 :      LIST
0000 303 :          BRW L40
0000 304 :      L20:
0000 305 :          CREMAPSEC STARTVA,ENDVA,STATUS=STAT,-
0000 306 :                   FLAGS=#<SECSM_GBL!SECSM_PERM!SECSM_DZRO!SECSM_WRT>,-
0000 307 :                   PAGCNT=#25
0000 308 :      LIST
0000 309 :          BRW L40
0000 310 :      L30:
0000 311 :          .IF      NB,STARTVA
0000 312 :          MOVL    STARTVA,W^INRANGE
0000 313 :          .ENDC
0000 314 :          .IF      NB,ENDVA
0000 315 :          MOVL    ENDVA,W^INRANGE+4
0000 316 :          .ENDC
0000 317 :          MOVZWL  STAT,R3
0000 318 :      L2:
0000 319 :          $CRMPSC_S
0000 320 :              -
0000 321 :              INADR=<W^INRANGE>,-
0000 322 :              RETADR=<W^RETRANGE>,-
0000 323 :              GSDNAM=<W^GBLSECNAM>,-
0000 324 :              VBN=#0,-
0000 325 :              FLAGS=#<SECSM_GBL!SECSM_PERM!SECSM_PFNMAP!SECSM_WRT>,-
0000 326 :              PAGCNT=#25
0000 327 :      *****
0000 328 :      :IN ORDER TO REMOVE THE TEMPORARY FIX EXPLAINED ABOVE REMOVE THE LINES
0000 329 :      :STARTING FROM HERE AND EXTENDING THE THE NEXT SET OF ASTERICS.
0000 330 :
0000 331 :          CMPL   #^X1828A,R0                ;IF PREVIOUS GBL SECTION HASN'T
0000 332 :          BEQL   L2                          ;BEEN DELETED TRY CREMAPSEC AGAIN.
0000 333 :      *****
0000 334 :          MOVAL  W^CRMPSCERR,R1
0000 335 :          BSBW  CHECK1
0000 336 :      L40:
0000 337 :          NLIST

```

```
0000 337 .ENDM MAKEVA
0000 338
0000 339 .MACRO RANGECHK ONOROFF
0000 340 LIST
0000 341 .IF IDN <ONOROFF>,<OFF>
0000 342 BICL #CTL$M_RNGCHK,W^CTLFLG
0000 343 .IFF
0000 344 BISL #CTL$M_RNGCHK,W^CTLFLG
0000 345 .ENDC
0000 346 NLIST
0000 347 .ENDM RANGECHK
0000 348
0000 349 :
0000 350 : EQUATED SYMBOLS:
0000 351 :
0000 352 $SECDDEF
0000 353 $SSDEF
0000 354 $PRTDEF
0000 355 $GBLINI
0000 356 $VIELD CTL,0,<-
0000 357 <MEMLOOP,,MASK>,-
0000 358 <TSTLOOP,,MASK>,-
0000 359 <PIDMSG,,MASK>,-
0000 360 <RNGCHK,,MASK>-
0000 361 >
00000010 0000 362 PRT$C NONE=104
00000002 0000 363 RUN2=2
00000001 0000 364 RUN1=1
00000003 0000 365 RUNMAX=3
0000 366 :
0000 367 : OWN STORAGE:
0000 368 :
```

```
;DEFINE CONTROL BITS IN R3
;LOOP IN MEMORY WRITE LOOP
;REDO ENTIRE TEST FROM TOP
;PUT PROCESS ID IN EACH TYPEOUT
;ON IF CHECKING RETURN RANGE
```

```
0000 370 .SBTTL DATA STORAGE AND MESSAGE STRINGS
00000000 371 .PSECT DATAO,PAGE,WRT,NOEXE
00000008 0000 372 INRANGE:
00000010 0008 373 .BLKL 2
00000006 0010 374 RETRANGE:
00000018 0014 375 .BLKL 2
0000001C 0018 376 CTLFLG: .LONG CTL$M_TSTLOOP!CTL$M_PIDMSG
00000020 001C 377 SAVEND: .BLKL 1
00000003 0020 378 HIGHPOADR:
00000028 0024 379 .BLKL 1 ;LAST BYTE ADDRESS IN PO SPACE
00000003 0020 380 PID: .BLKL 1 ;PROCESS ID
00000028 0024 381 MAXPASSCNT:
00000003 0020 382 .LONG 3 ;NUMBER OF PASSES TO RUN
00000028 0024 383 PASSCNT:
00000003 0028 384 .BLKL 1 ;PASS COUNTER
00000030 002C 385 MAXWHICHRUN:
00000034 0030 386 .LONG 3 ;NUMBER OF RUNS
0000004C 0034 387 WHICHRUN:
0000003C 004C 388 .BLKL 1 ;WHICH RUN COUNTER
00000044 004C 389 WRKSETLIM:
00000050 004C 390 .BLKL 1 ;RETURNED NEW WORKING SET LIMIT
0000004C 0034 391 WRKSETDEF:
0000003C 004C 392 .BLKL 6 ;DEFAULT, MAX, MIN WORKING SET LIMIT
00000044 004C 393 WRKSETMAX=WRKSETDEF+8
00000050 004C 394 WRKSETMIN=WRKSETDEF+16
00000050 004C 395 WRKSETMAXADD:
00000050 0050 396 .BLKL 1 ;WRKSETMAX-WRKSETDEF
00000050 0050 397 PREVPROT:
000000E8 00E4 398 FAB: $FAB FAC=PUT, FNA=OUTNAMADR, FNS=OUTNAMSIZ ;FAB FOR OUTPUT
00000106'000000A0' 00FA 399 RAB: $RAB FAB=FAB ;RECORD ACCESS BLOCK FOR OUTPUT
00000102'00000004' 00F0 400 MSGLEN: .BLKL 1 ;RETURN LENGTH FROM FAO
00000102'00000004' 00F0 401 MSGBUFD: .LONG MSGBUFSIZ,MSGBUF ;MESSAGE BUFFER DESCRIPTOR
00000102'00000004' 00F0 402 PIDMSGD:
00000102'00000004' 00F0 403 .LONG MSGBUF-PIDMSG,PIDMSG
00000102'00000004' 00F8 404 :
00000102'00000004' 00F8 405 : ***** DO NOT SEPARATE OR REORDER THE FOLLOWING LINES
00000102'00000004' 00F8 406 :
00000102'00000004' 00F8 407 MSGBUFID:
00000102'00000004' 00F8 408 CRLF: .BYTE ^015,^012
20 53 53 45 43 4F 52 50 00FA 409 .ASCII $PROCESS $
20 20 20 20 0102 410 PIDMSG: .ASCII $ $
000001A6 0106 411 MSGBUF: .BLKB 160 ;MESSAGE BUFFER USED BY FAO
000001A6 01A6 412 MSGBUFSIZ=-MSGBUF
000001A6 01A6 413 :
000001A6 01A6 414 : ***** DO NOT SEPARATE OR REORDER THE PRECEEDING LINES
000001A6 01A6 415 :
000001A6 01A6 416 :
```

												00000000	418		.PSECT CODE,PAGE,NOWRT,EXE
												0000	419		
												0000	420		OUTNAMADR:
54	55	50	54	55	4F	24	53	59	53			0000	421		.ASCII /SYS\$OUTPUT/
											0000000A	000A	422		OUTNAMSIZ=-OUTNAMADR
												000A	423		
												000A	424		CRETVAERRADR:
52	52	45	20	41	56	54	45	52	43	2F	21	000A	425		.ASCII \$!/CRETVA ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL\$
58	21	20	3D	20	43	50	20	2D	20	52	4F	0016			
41	57	20	53	55	54	41	54	53	20	2C	4C	0022			
4C	55	4F	48	53	20	2C	4C	58	21	20	53	002E			
												003A			
21	20	3D	20	52	44	41	4E	49	09	2F	21	0042	426		.ASCII \$!/ INADR = !XL - !XL, RETADR = !XL - !XL!/\$
52	20	20	2C	4C	58	21	20	2D	20	4C	58	004E			
20	4C	58	21	20	3D	20	52	44	41	54	45	005A			
												0066			
												00000063	427		CRETVAERRSIZ=-CRETVAERRADR
												006D	428		
												006D	429		DELTVAERRADR:
52	52	45	20	41	56	54	4C	45	44	2F	21	006D	430		.ASCII \$!/DELTVA ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL\$
58	21	20	3D	20	43	50	20	2D	20	52	4F	0079			
41	57	20	53	55	54	41	54	53	20	2C	4C	0085			
4C	55	4F	48	53	20	2C	4C	58	21	20	53	0091			
												009D			
21	20	3D	20	52	44	41	4E	49	09	2F	21	00A5	431		.ASCII \$!/ INADR = !XL - !XL, RETADR = !XL - !XL!/\$
52	20	20	2C	4C	58	21	20	2D	20	4C	58	00B1			
20	4C	58	21	20	3D	20	52	44	41	54	45	00BD			
												00C9			
												00000063	432		DELTVAERRSIZ=-DELTVAERRADR
												00D0	433		
												00D0	434		CRMPSCERRADR:
52	52	45	20	43	53	50	4D	52	43	2F	21	00D0	435		.ASCII \$!/CRMPSC ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL\$
58	21	20	3D	20	43	50	20	2D	20	52	4F	00DC			
41	57	20	53	55	54	41	54	53	20	2C	4C	00E8			
4C	55	4F	48	53	20	2C	4C	58	21	20	53	00F4			
												0100			
21	20	3D	20	52	44	41	4E	49	09	2F	21	0108	436		.ASCII \$!/ INADR = !XL - !XL, RETADR = !XL - !XL!/\$
52	20	20	2C	4C	58	21	20	2D	20	4C	58	0114			
20	4C	58	21	20	3D	20	52	44	41	54	45	0120			
												012C			
												00000063	437		CRMPSCERRSIZ=-CRMPSCERRADR
												0133	438		
												0133	439		LKWSETERRADR:
52	52	45	20	54	45	53	57	4B	4C	2F	21	0133	440		.ASCII \$!/LKWSET ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL\$
58	21	20	3D	20	43	50	20	2D	20	52	4F	013F			
41	57	20	53	55	54	41	54	53	20	2C	4C	0148			
4C	55	4F	48	53	20	2C	4C	58	21	20	53	0157			
												0163			
21	20	3D	20	52	44	41	4E	49	09	2F	21	0168	441		.ASCII \$!/ INADR = !XL - !XL, RETADR = !XL - !XL!/\$
52	20	20	2C	4C	58	21	20	2D	20	4C	58	0177			
20	4C	58	21	20	3D	20	52	44	41	54	45	0183			
												018F			
												00000063	442		LKWSETERRSIZ=-LKWSETERRADR
												0196	443		
												0196	444		ULWSETERRADR:
52	52	45	20	54	45	53	57	4C	55	2F	21	0196	445		.ASCII \$!/ULWSET ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL\$
58	21	20	3D	20	43	50	20	2D	20	52	4F	01A2			

```

41 57 20 53 55 54 41 54 53 20 2C 4C 01AE
4C 55 4F 48 53 20 2C 4C 58 21 20 53 01BA
      4C 58 21 20 45 42 20 44 01C6
21 20 3D 20 52 44 41 4E 49 09 2F 21 01CE 446
52 20 20 2C 4C 58 21 20 2D 20 4C 58 01DA
20 4C 58 21 20 3D 20 52 44 41 54 45 01E6
      2F 21 4C 58 21 20 2D 01F2
      00000063 01F9 447
      01F9 448 ULWSETERRSIZ=-ULWSETERRADR
      01F9 449
41 57 20 53 55 54 41 54 53 20 2C 4C 0211 450
4C 55 4F 48 53 20 2C 4C 58 21 20 53 021D CNTREGERRADR:
      4C 58 21 20 45 42 20 44 0229 .51
20 3D 20 54 4E 43 47 41 50 09 2F 21 0231 .ASCII $!/ PAGCNT = !UL, REGION = P!UB SPACE, $
20 4E 4F 49 47 45 52 20 2C 4C 55 21 023D
45 43 41 50 53 20 42 55 21 50 20 3D 0249
      20 2C 0255
4C 58 21 20 3D 20 52 44 41 54 45 52 0257 452
      2F 21 4C 58 21 20 2D 20 0263 .ASCII $RETADR = !XL - !XL!/$
      00000072 026B 453
      026B 454 CNTREGERRSIZ=-CNTREGERRADR
      026B 455
41 57 20 53 55 54 41 54 53 20 2C 4C 0283 456
4C 55 4F 48 53 20 2C 4C 58 21 20 53 028F EXPREGERRADR:
      4C 58 21 20 45 42 20 44 029B .ASCII $!/EXPREG ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
20 3D 20 54 4E 43 47 41 50 09 2F 21 02A3 457
20 4E 4F 49 47 45 52 20 2C 4C 53 21 02AF .ASCII $!/ PAGCNT = !SL, REGION = P!UB SPACE, $
45 43 41 50 53 20 42 55 21 50 20 3D 02BB
      20 2C 02C7
4C 58 21 20 3D 20 52 44 41 54 45 52 02C9 458
      2F 21 4C 58 21 20 2D 20 02D5 .ASCII $RETADR = !XL - !XL!/$
      00000072 02DD 459
      02DD 460 EXPREGERRSIZ=-EXPREGERRADR
      02DD 461
41 57 20 53 55 54 41 54 53 20 2C 4C 02E9 462
4C 55 4F 48 53 20 2C 4C 58 21 20 53 41 0301 ADJWSLERRADR:
      4C 58 21 20 45 42 20 44 4C 030D .ASCII $!/ADJWSL ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
20 3D 20 54 4E 43 47 41 50 09 2F 21 0316 463
45 43 41 50 53 20 42 55 21 50 20 3D 0322 .ASCII $!/ PAGCNT = !SL, LIMIT WAS !UW, SHOULD BE !UW!/$
45 4F 48 53 20 2C 57 55 21 20 53 41 032E
      2F 21 57 55 21 20 45 42 20 44 4C 033A
      00000068 0345 464
      0345 465 ADJWSLERRSIZ=-ADJWSLERRADR
      0345 466
45 53 20 47 4E 49 48 52 4F 57 2F 21 0345 467 WSETLMCTLADR:
45 4B 43 4F 4C 2F 45 5A 49 53 20 54 0351 .ASCII $!/WORKING SET SIZE/LOCKED LIMITS: $
      20 20 3A 53 54 49 4D 49 4C 20 44 035D
2F 4C 55 21 20 54 4C 55 41 46 45 44 0368 468
4C 55 21 20 58 41 4D 20 2C 4C 55 21 0374 .ASCII $DEFAULT !UL!/UL, MAX !UL!/UL, MIN !UL!/UL!/$
55 21 20 4E 49 4D 20 2C 4C 55 21 2F 0380
      2F 21 4C 55 21 2F 4C 038C

```

```

0000004E 0393 469 WSETLMCTLSIZ=-WSETLMCTLADR
0393 470
0393 471 READERRADR:
4F 52 52 45 20 44 41 45 52 20 2F 21 0393 472 .ASCII $!/ READ ERROR - LOCATION = !XL !/$
4E 4F 49 54 41 43 4F 4C 20 2D 20 52 039F
2F 21 20 4C 58 21 20 3D 20 03AB
00000021 03B4 473 READERRSIZ=-READERRADR
03B4 474
03B4 475 NOREADERRADR:
45 20 44 41 45 52 2D 4F 4E 20 2F 21 03B4 476 .ASCII $!/ NO-READ ERROR - LOCATION = !XL !/$
54 41 43 4F 4C 20 2D 20 52 4F 52 52 03C0
2F 21 20 4C 58 21 20 3D 20 4E 4F 49 03CC
00000024 03D8 477 NOREADERRSIZ=-NOREADERRADR
03D8 478
03D8 479 WRITERRADR:
52 52 45 20 45 54 49 52 57 20 2F 21 03D8 480 .ASCII $!/ WRITE ERROR - LOCATION = !XL !/$
4F 49 54 41 43 4F 4C 20 2D 20 52 4F 03E4
2F 21 20 4C 58 21 20 3D 20 4E 03F0
00000022 03FA 481 WRITERRSIZ=-WRITERRADR
03FA 482
03FA 483 NOWRITERRADR:
20 45 54 49 52 57 2D 4F 4E 20 2F 21 03FA 484 .ASCII $!/ NO-WRITE ERROR - LOCATION = !XL !/$
41 43 4F 4C 20 2D 20 52 4F 52 52 45 0406
21 20 4C 58 21 20 3D 20 4E 4F 49 54 0412
2F 041E
00000025 041F 485 NOWRITERRSIZ=-NOWRITERRADR
041F 486
041F 487 MEMLOOPCTLADR:
4F 4F 4C 20 59 52 4F 4D 45 4D 2F 21 041F 488 .ASCII $!/MEMORY LOOP ERRORS
042B
43 4F 4C 42 20 45 4C 49 46 09 2F 21 0432 489 .ASCII $!/ FILE BLOCK WAS !XL, SHOULD BE !XLS
53 20 2C 4C 58 21 20 53 41 57 09 4B 043E
4C 58 21 20 45 42 20 44 4C 55 4F 48 044A
44 41 20 59 52 4F 4D 45 4D 09 2F 21 0456 490 .ASCII $!/ MEMORY ADR WAS !XL, SHOULD BE !XLS
53 20 2C 4C 58 21 20 53 41 57 09 52 0462
4C 58 21 20 45 42 20 44 4C 55 4F 48 046E
49 20 53 53 45 43 4F 52 50 09 2F 21 047A 491 .ASCII $!/ PROCESS ID WAS !XL, SHOULD BE !XLS
53 20 2C 4C 58 21 20 53 41 57 09 44 0486
4C 58 21 20 45 42 20 44 4C 55 4F 48 0492
0000007F 049E 492 MEMLOOPCTLSIZ=-MEMLOOPCTLADR
049E 493
049E 494 RANGERRADR:
4E 41 52 20 4E 52 55 54 45 52 2F 21 049E 495 .ASCII $!/RETURN RANGE ERROR - LOCATION = !XLS
4C 20 2D 20 52 4F 52 52 45 20 45 47 04AA
58 21 20 3D 20 4E 4F 49 54 41 43 4F 04B6
4C
21 20 3D 20 52 44 41 4E 49 09 2F 21 04C3 496 .ASCII $!/ 'NADR = !XL - !XL, RETADR = !XL - !XL!/$
45 52 20 2C 4C 58 21 20 2D 20 4C 58 04CF
2D 20 4C 58 21 20 3D 20 52 44 41 54 04DB
2F 21 4C 58 21 20 04E7
0000004F 04ED 497 RANGERRSIZ=-RANGERRADR
04ED 498
04ED 499 IDMSGADR:
4E 41 4D 20 59 52 4F 4D 45 4D 2F 21 04ED 500 .ASCII $!/MEMORY MANAGEMENT SERVICES TEST #7 (XQUOTA), PASS !UL!/$
56 52 45 53 20 54 4E 45 4D 45 47 41 04F9
37 23 20 54 53 45 54 20 53 45 43 49 0505
50 20 2C 29 41 54 4F 55 51 58 28 20 0511

```

```

2F 21 4C 55 21 20 53 53 41 051D
00000039 0526
0526 501
C526 502
0526 503 RUN1_MSGADR:
0532 504 .ASCII $!/ ***** TEST WILL NOW BE RUN USING REGULAR VA SPACE *****$
053E
054A
0556
0562
0566 505 .ASCII $!/ $
0569 506 RUN1_MSGSIZ=-.RUN1_MSGADR
0569 507
0569 508 RUN2_MSGADR:
0575 509 .ASCII $!/ ***** TEST WILL NOW BE RUN USING PERM GBL SECTIONS *****$
0581
058D
0599
05A5
05AA 510 .ASCII $!/ $
05AD 511 RUN2_MSGSIZ=-.RUN2_MSGADR
05AD 512
05AD 513 RUN3_MSGADR:
05B9 514 .ASCII $!/ ***** TEST WILL NOW BE RUN USING PFN MAPPING *****$
05C5
05D1
05DD
05E8 515 .ASCII $!/ $
05EB 516 RUN3_MSGSIZ=-.RUN3_MSGADR
05EB 517
05EB 518 PIDCTLADR:
05EB 519 .ASCII $!UL$
05EE 520 PIDCTLSIZ=-.PIDCTLADR
05EE 521
05EE 522 GBLSECNAMADR:
05EE 523 .ASCII $MMGTST$
05F4 524 GBLSECNAMSIZ=-.GBLSECNAMADR
05F4 525
05F4 526 FILENAMEADR:
05F4 527 .ASCII $MMGTST.DAT$
05FE 528 FILENAMESIZ=-.FILENAMEADR
05FE 529

```

```

501 IDMSGsiz=-.IDMSGADR
502
503 RUN1_MSGADR:
504 .ASCII $!/ ***** TEST WILL NOW BE RUN USING REGULAR VA SPACE *****$
505 .ASCII $!/ $
506 RUN1_MSGSIZ=-.RUN1_MSGADR
507
508 RUN2_MSGADR:
509 .ASCII $!/ ***** TEST WILL NOW BE RUN USING PERM GBL SECTIONS *****$
510 .ASCII $!/ $
511 RUN2_MSGSIZ=-.RUN2_MSGADR
512
513 RUN3_MSGADR:
514 .ASCII $!/ ***** TEST WILL NOW BE RUN USING PFN MAPPING *****$
515 .ASCII $!/ $
516 RUN3_MSGSIZ=-.RUN3_MSGADR
517
518 PIDCTLADR:
519 .ASCII $!UL$
520 PIDCTLSIZ=-.PIDCTLADR
521
522 GBLSECNAMADR:
523 .ASCII $MMGTST$
524 GBLSECNAMSIZ=-.GBLSECNAMADR
525
526 FILENAMEADR:
527 .ASCII $MMGTST.DAT$
528 FILENAMESIZ=-.FILENAMEADR
529

```

```

05FE 531 :
05FE 532 : STRING DESCRIPTORS
05FE 533 :
05FE 534 .ALIGN LONG
0600 535
0600 536 CRETVAERR:
0000000A'00000063 0600 537 .LONG CRETVAERRSIZ,CRETVAERRADR
0608 538 DELTVAERR:
0000006D'00000063 0608 539 .LONG DELTVAERRSIZ,DELTVAERRADR
0610 540 CNTREGERR:
000001F9'00000072 0610 541 .LONG CNTREGERRSIZ,CNTREGERRADR
0618 542 CRMPSCERR:
000000D0'00000063 0618 543 .LONG CRMPSCERRSIZ,CRMPSCERRADR
0620 544 EXPREGERR:
0000026B'00000072 0620 545 .LONG EXPREGERRSIZ,EXPREGERRADR
0628 546 LKWSSETERR:
00000133'00000063 0628 547 .LONG LKWSSETERRSIZ,LKWSSETERRADR
0630 548 ULWSSETERR:
00000196'00000063 0630 549 .LONG ULWSSETERRSIZ,ULWSSETERRADR
0638 550 ADJWSLERR:
000002DD'00000068 0638 551 .LONG ADJWSLERRSIZ,ADJWSLERRADR
0640 552 WSETLMCTL:
00000345'0000004E 0640 553 .LONG WSETLMCTLSIZ,WSETLMCTLADR
0648 554 READERR:
00000393'00000021 0648 555 .LONG READERRSIZ,READERRADR
0650 556 NOREADERR:
000003B4'00000024 0650 557 .LONG NOREADERRSIZ,NOREADERRADR
0658 558 WRITERR:
000003D8'00000022 0658 559 .LONG WRITERRSIZ,WRITERRADR
0660 560 NOWRITERR:
000003FA'00000025 0660 561 .LONG NOWRITERRSIZ,NOWRITERRADR
0668 562 MEMLOOPCTL:
0000041F'0000007F 0668 563 .LONG MEMLOOPCTLSIZ,MEMLOOPCTLADR
0670 564 RANGERR:
0000049E'0000004F 0670 565 .LONG RANGERRSIZ,RANGERRADR
0678 566 IDMSG:
000004ED'00000039 0678 567 .LONG IDMSGsiz,IDMSGADR
0680 568 RUN1_MSG:
00000526'00000043 0680 569 .LONG RUN1_MSGSIZ,RUN1_MSGADR
0688 570 RUN2_MSG:
00000569'00000044 0688 571 .LONG RUN2_MSGSIZ,RUN2_MSGADR
0690 572 RUN3_MSG:
000005AD'0000003E 0690 573 .LONG RUN3_MSGSIZ,RUN3_MSGADR
0698 574 PIDCTL:
000005EB'00000003 0698 575 .LONG PIDCTLSIZ,PIDCTLADR
06A0 576 GBLSECNAM:
000005EE'00000006 06A0 577 .LONG GBLSECNAMSIZ,GBLSECNAMADR
06A8 578 FILENAME:
000005F4'0000000A 06A8 579 .LONG FILENAMESIZ,FILENAMEADR
06B0 580

```



```

06B0 582 .SBTTL INITIALIZATION
06B0 583 :*****
06B0 584 :PROGRAM DESCRIPTION:
06B0 585 :
06B0 586 :     THIS PROGRAM TESTS THE FOLLOWING SYSTEM SERVICES:
06B0 587 :         $CRETVA, $SEXPREG
06B0 588 :
06B0 589 :     THE SYSTEM SERVICES ARE TESTED IN THE FOLLOWING MANNER. THE PROGRAM
06B0 590 :     IS DESIGNED TO MAKE THREE RUNS. EACH RUN MAKES THREE PASSES. ON
06B0 591 :     THE FIRST RUN WHEN A MAKEVA IS ENCOUNTERED IT IS REPLACED WITH A
06B0 592 :     CRETVA. IT THEN MAKES THREE PASSES THROUGH THE PROGRAM DOING
06B0 593 :     ALL THE POSSIBLE TESTS. ON THE SECOND RUN THE MAKEVA IS ENCOUNTERED
06B0 594 :     AND IS REPLACED WITH A $CRMPSC THAT MAPS A PERMANENT GLOBAL SECTION.
06B0 595 :     FOR THE THIRD RUN, THE MAKVA IS REPLACED WITH A $CRMPSC THAT MAPS
06B0 596 :     A PFN GLOBAL SECTION. USING THIS APPROACH THE $CRMPSC SYSTEM SERVICE
06B0 597 :     WAS ALSO ABLE TO BE TESTED AS ALL ERROR PATHS FOLLOWED WHEN USING
06B0 598 :     NORMAL VA ARE ALSO APPLICABLE WHEN USING GBL SECTION MAPPING.
06B0 599 :     FOR EACH OF THESE RUNS A MESSAGE IS SENT TO THE TERMINAL
06B0 600 :     INFORMING THE OPERATOR AS TO WHICH OF THESE SUBTITUTE MAKEVA'S WILL
06B0 601 :     BE USED FOR THAT SET OF PASSES.
06B0 602 :
06B0 603 :     REFER TO MASD$:[MMGTST.COM]MMGTST.RAP FOR FURTHER INFORMATION
06B0 604 :     REGARDING JUST HOW COMPLETELY THE ABOVE MENTIONED SYSTEM SERVICES
06B0 605 :     ARE TESTED BY THIS PROGRAM.
06B0 606 :
06B0 607 : *NOTE:
06B0 608 :     THERE IS A BUG IN $CRMPSC THAT WILL BE FIXED IN THE FUTURE. UNTIL
06B0 609 :     THIS FIX TAKES PLACE THERE AREA TEMPORARY PATCH'S LOCATED AT THE END
06B0 610 :     OF THE CREMAPSEC MACRO AND MAKVA MACRO WHICH CIRCUMVENT THIS
06B0 611 :     PROBLEM. UPON CORRECTION OF THIS BUG REFER TO THESE ABOVE MENTIONED
06B0 612 :     MACROS TO FIND OUT HOW TO REMOVE THESE TEMPORARY PATCHES.
06B0 613 :     ALSO, THE FILE MMGTST.DAT, WHICH IS CREATED BY RUNNING MMGCRTFIL.MAR,
06B0 614 :     IS NECESSARY TO RUN THIS PROGRAM AS IT IS REFERENCED BY CREMAPSEC.
06B0 615 :
06B0 616 : *PRIVILEGES:
06B0 617 :     IN ORDER TO RUN THIS PROGRAM YOU MUST HAVE IN YOUR POSSESSION
06B0 618 :     THE PRIVILEGES PRMGBL AND PFNMAP.
06B0 619 : *****
06B0 620 :
06B0 621 : START HERE
06B0 622 :
0000 06B0 623 START: .WORD 0 ;ENTRY MASK
OE 50 E9 06B2 624 $OPEN W^FAB ;OPEN THE FILE '$OUTPUT'
06C0 625 BLBC RO,10$ ;BRANCH IF ERROR
09 50 E8 06CB 626 $CONNECT W^RAB ;CONNECT THE RECORD ACCESS BLOCK
06CE 627 BLBS RO,20$
06D7 628 10$: $EXIT S RO ;EXIT WITH STATUS IN RO
50 000001C'EF 3C 06E6 629 20$: $RESUME_S PID ;SET UP PROCESS ID
06ED 630 MOVZWL PID,RO
0705 631 $FAO_S PIDCTL,MSGLEN,PIDMSGD,RO ;INIT THE PROCESS ID STRING
070B 632 CLRL WHICHRUN ;INIT THE RUN COUNT TO ZERO
070B 633
070B 634
070B 635 MAIN PROGRAM:
070B 636 :*****
070B 637 :     THIS COUNTER IS USED TO DETERMINE WHICH CRMPSC IS TO BE
070B 638 :     REFERENCED, OR IF THE CRETVA IS TO BE USED, WHEN A MAKEVA

```

```

070B 639 : IS ENCOUNTERED.
070B 640 :*****
070B 641
OC 0024'CF 01 DO 070B 642      MOVL #1,W^PASSCNT      ;INITIALIZE THE PASS COUNT
002C'CF 0028'CF F3 0710 643      AOBLEQ W^MAXWHICHRUN,W^WHICHRUN,PRESTART
0718 644
0718 645 END:
0718 646      MOVZBL #SS$_NORMAL,R0
071B 647      $EXIT,S R0
0724 648
03 01 002C'CF CF 0724 649 PRESTART:
0724 650      CASEL W^WHICHRUN,#1,#RUNMAX
072A 651 L50:
0006' 072A 652      .WORD CRETVA_MESSAGE-L50      ;FOR EACH RUN PRINT MESSAGE
0022' 072C 653      .WORD GBL_SEC_MESSAGE-L50      ;DESCRIBING WHICH VA SPACE
0043' 072E 654      .WORD PFNMAP_MESSAGE-L50      ;IS BEING USED.
0730 655
0730 656 CRETVA_MESSAGE:
0730 657      $FAO,S RUN1 MSG,MSGLEN,MSGBUFD
03F5 30 0747 658      BSBW- TYPEMSGBUF
40 11 074A 659      BRB RSTART
074C 660
0010'CF 04 C8 074C 661 GBL_SEC_MESSAGE:
074C 662      BISL #CTLSM_PIDMSG,W^CTLFLG      ;ALLOW PROCESS ID TO PRINT
0751 663      $FAO,S RUN2 MSG,MSGLEN,MSGBUFD
03D4 30 0768 664      BSBW- TYPEMSGBUF
1F 11 076B 665      BRB RSTART
076D 666
0010'CF 04 C8 076D 667 PFNMAP_MESSAGE:
076D 668      BISL #CTLSM_PIDMSG,W^CTLFLG      ;ALLOW PROCESS ID TO PRINT
0772 669      $FAO,S RUN3 MSG,MSGLEN,MSGBUFD
03B3 30 0789 670      BSBW- TYPEMSGBUF
078C 671
078C 672 RSTART:
078C 673      RANGECHK ON
0010'CF 08 C8 078C 674      BISL #CTLSM_RNGCHK,W^CTLFLG
0010'CF 04 CA 0791 674      BICL #CTLSM_PIDMSG,W^CTLFLG      ;STOP PROCESS ID FROM PRINTING
0796 675      $FAO,S IDMSG,MSGLEN,MSGBUFD,PASSCNT
0389 30 07B3 676      BSBW- TYPEMSGBUF
07B6 677      EXPREG #1
53 01 3C 07B6 678      MOVZWL S^#SS$_NORMAL,R3
54 01 DO 07B9 678      MOVL #1,R4
51 0008'CF DE 07BC 679      MOVAL W^RETRANGE,R1
55 D4 07C1 679      CLRL R5
0243 30 07C3 679      BSBW EXPREGSUBR
52 0008'CF 7D 07C6 678      MOVQ W^RETRANGE,R2
0000'CF 52 7D 07CB 679      MOVQ R2,W^INRANGE
0014'CF 52 DO 07D0 680      MOVL R2,W^SAVEND
07D5 681      DELTVA
01 002C'CF B1 07D5 681      CMPW W^WHICHRUN,#RUN1
07DA 681      BLEQ 30000$
07DC 681      PUSHL #0
FEBE CF 7F 07DE 681      PUSHAQ W^GBLSECNAM
07E2 681      PUSHL #0
00000000'GF 03 FB 07E4 681      CALLS #3,G^SYSDGBLSC
07EB 681
53 01 3C 07EB 681      MOVZWL S^#SS$_NORMAL,R3

```

MMGXQUOTA
V04-000

N 14
- TEST MMG SERVICES WITH LOW PGFLQUOTA
INITIALIZATION

16-SEP-1984 02:05:45
5-SEP-1984 01:58:27

VAX/VMS Macro V04-00
[MMGTST.SRC]MMGXQUOTA.MAR;1

Page 16
(6)

50	0000'CF	DE	07EE	
51	0008'CF	DE	07F3	
	015F	30	07FB	
			07FB	682
			07FB	683

MOVAL	W^INRANGE,R0
MOVAL	W^RETRANGÉ,R1
BSBW	DELTVASUBR

```

07FB 685 .SBTTL FORCE ERRORS IN CRETVA
07FB 686 :
07FB 687 : FORCE ERRORS FROM CRETVA
07FB 688 :
07FB 689 :*****
07FB 690 : THIS MAKEVA IS NOT AVAILABLE FOR THE SECOND RUN, WHEN PERM
07FB 691 : GLOBAL SECTIONS ARE USED, AS IN ORDER FOR THE EXPECTED ERROR TO
07FB 692 : BE RETURNED THE PAGES MUST BE CRF AND SINCE SHARED MEMORY TESTING
07FB 693 : IS POSSIBLE WITH THIS TEST (AND CRF IS NOT PERMITTED WITH SHARED
07FB 694 : MEMORY) THIS SECTION WAS BYPASSED FOR THAT REASON.
07FB 695 :*****
07FB 696 :
02 002C'CF B1 07FB 697 CMPW W^WHICHRUN,#RUN2
03 03 12 0800 BNEQ 10$
00F2 31 0802 BRW REROUTE
0805 700 10$:
0805 701
0805 702 MAKEVA W^SAVEND,#1@30-1,#SS$ EXQUOTA ;EXCEED PAGE FILE QUOTA
03 01 002C'CF CF 0805 CASEL W^WHICHRUN,#1,#RUNMAX
0808 30001$:
0006' 0808 .WORD 30003$-30001$ ;IF FIRST RUN USE CRETVA.
0029' 080D .WORD 30004$-30001$ ;IF SECOND USE GBL CREMAPSEC
0073' 080F .WORD 30005$-30001$ ;IF THIRD USE PFN MAPPING
0811 30003$:
0000'CF 0014'CF DO 0811 MOVL W^SAVEND,W^INRANGE
0004'CF 3FFFFFFF 8F DO 0818 MOVL #1@30-1,W^INRANGE+4
53 1C 3C 0821 MOVZWL #SS$ EXQUOTA,R3
50 0000'CF DE 0824 MOVAL W^INRANGE,R0
51 0008'CF DE 0829 MOVAL W^RETRANGE,R1
0115 30 082E BSBW CRETVASUBR
0096 31 0831 BRW 30006$
0834 30004$:
0000'CF 0014'CF DO 0834 MOVL W^SAVEND,W^INRANGE
0004'CF 3FFFFFFF 8F DO 0838 MOVL #1@30-1,W^INRANGE+4
53 1C 3C 0844 MOVZWL #SS$ EXQUOTA,R3
0847 30007$:
00 DD 0847 PUSHL #0
19 DD 0849 PUSHL #25
FE59 CF DF 0848 PUSHAL W^FILENAME
00 DD 084F PUSHL #0
FE4B CF DF 0851 PUSHAL W^GBLSECNAM
0000400D 8F DD 0855 PUSHL #<SECSM_GBL!SECSM_PERM!SECSM_DZRO!SECSM_WRT>
0008'CF DF 0858 PUSHAL W^RETRANGE
0000'CF DF 085F PUSHAL W^INRANGE
00000000'EF 08 FB 0863 CALLS #8,LIB$ CREMAPSEC
50 0001828A 8F D1 086A CMPL #^X1828A,R0 ;IF PREVIOUS GBL SECTION HASN'T
D4 13 0871 BEQL 30007$ ;BEEN DELETED TRY CREMAPSEC AGAIN.
51 FDA1 CF DE 0873 MOVAL W^CRMPSCERR,R1
0119 30 0878 BSBW CHECK1
004C 31 087B BRW 30006$
087E 30005$:
0000'CF 0014'CF DO 087E MOVL W^SAVEND,W^INRANGE
0004'CF 3FFFFFFF 8F DO 0885 MOVL #1@30-1,W^INRANGE+4
53 1C 3C 088E MOVZWL #SS$ EXQUOTA,R3
0891 30002$:
7E 7C 0891 CLRQ -(SP)
00 DD 0893 PUSHL #0

```



```

08F7 708 .SBTTL FORCE ERRORS FROM EXPREG
08F7 709 :
08F7 710 : FORCE ERRORS FROM EXPREG
08F7 711 :
08F7 712 :
EXPREG #1@21-1,#0,STATUS=#SS$ EXQUOTA ;EXCEED PAGE FILE QUOTA
MOVZWL #SS$ EXQUOTA,R3
MOVL #1@21-1,R4
MOVAL W^RETRANGE,R1
CLRL R5
BSBW EXPREGSUBR
W^RETRANGE,W^INRANGE
MOVQ DELTVA
DELTVA
CMPW W^WHICHRUN,#RUN1 ;DELETE WHAT WE CREATED
BLEQ 30009$ ;IF USING MAPPED SECTIONS FOR TESTS,
PUSH #0 ;DELETE SECTION AND THEN DO A DELTVA
PUSHAQ W^GBLSECNAM
PUSHL #0
CALLS #3,G^SYS$DGBLSC
30009$:
MOVZWL S^#SS$ NORMAL,R3
MOVAL W^INRANGE,R0
MOVAL W^RETRANGE,R1
BSBW DELTVASUBR
0938 715 :
0938 716 :
0938 717 :END OF LOOP
0938 718 :
0938 719 :
0938 719 AOBLEQ W^MAXPASSCNT,W^PASSCNT,160$
0940 720 150$: BRW MAIN_PROGRAM
0943 721 :
0943 722 160$: BRW RSTART

```

54 001FFFFF 8F 3C 08F7 53 1C 3C 08F7
51 0008'CF DE 08FA 00FE 55 D0 08FA
0000'CF 0008'CF D4 0901 00FE 55 D4 0906
01 002C'CF B1 0908 00FE 55 D4 0908
0F 15 0912 00FE 55 D4 0912
FD81 CF 7F 0917 00FE 55 D4 0919
00 DD 0919 00FE 55 D4 0919
00000000'GF 03 FB 091B 00FE 55 D4 091B
00 DD 091B 00FE 55 D4 091B
53 01 3C 0921 00FE 55 D4 0921
50 0000'CF DE 0928 00FE 55 D4 0928
51 0008'CF DE 0928 00FE 55 D4 0928
0022 30 0928 00FE 55 D4 0928
0938 715 :
0938 716 :
0938 717 :END OF LOOP
0938 718 :
03 0024'CF 0020'CF F3 0938 719 AOBLEQ W^MAXPASSCNT,W^PASSCNT,160\$
FDC8 31 0940 720 150\$: BRW MAIN_PROGRAM
FE46 31 0943 721 :
0943 722 160\$: BRW RSTART

-S
DE
LII

```
0946 724 .SBTTL SUBROUTINES TO CALL THE SERVICES
0946 725 :
0946 726 : INPUT:
0946 727 :
0946 728 : R0 = INADR
0946 729 : R1 = RETADR
0946 730 : R3 = DESIRED STATUS
0946 731 :
0946 732 : OUTPUT:
0946 733 :
0946 734 : R2 PRESERVED
0946 735 :
0946 736 CRETVASUBR:
0946 737 $CRETVA_S (R0),(R1)
51 FCA9 CF DE 0953 738 MOVAL -W^CRETVAERR,R1 ;ERROR CONTROL STRING
3A 11 0958 739 BRB CHECK1
095A 740 :
095A 741 : INPUT:
095A 742 :
095A 743 : R0 = INADR
095A 744 : R1 = RETADR
095A 745 : R3 = DESIRED STATUS
095A 746 :
095A 747 : OUTPUT:
095A 748 :
095A 749 : R2 PRESERVED
095A 750 :
095A 751 DELTVASUBR:
51 FC9D CF DE 095A 752 $DELTVA_S (R0),(R1) ;ERROR CONTROL STRING
26 11 0967 753 MOVAL -W^DELTVAERR,R1
096C 754 BRB CHECK1
096E 755 :
096E 756 : INPUT:
096E 757 :
096E 758 : R0 = INADR
096E 759 : R1 = RETADR
096E 760 : R3 = DESIRED STATUS
096E 761 :
096E 762 : OUTPUT:
096E 763 :
096E 764 : R2 PRESERVED
096E 765 :
096E 766 LKWSETSUBR:
51 FCA9 CF DE 096E 767 $LKWSET_S (R0),(R1) ;ERROR CONTROL STRING
12 11 097B 768 MOVAL -W^LKWSETERR,R1
0980 769 BRB CHECK1
0982 770 :
0982 771 : INPUT:
0982 772 :
0982 773 : R0 = INADR
0982 774 : R1 = RETADR
0982 775 : R3 = DESIRED STATUS
0982 776 :
0982 777 : OUTPUT:
0982 778 :
0982 779 : R2 PRESERVED
0982 780 :
```

-S

Ps

--

SP

MO

MO

NP

NP

SG

SO

NP

SC

```

0982 781 ULWSETSUBR:
0982 782 $ULWSET_S (R0), (R1)
51 FC9D CF DE 098F 783 MOVAL -W^ULWSETERR,R1 ;ERROR CONTROL STRING
0994 784 CHECK1:
53 50 D1 0994 785 CMPL R0,R3 ;STATUS AS DESIRED
57 13 0997 786 BEQL 10$ ;BRANCH IF YES
53 0244 8F B1 0999 787 CMPW #SS$_VASFULL,R3 ;IF EXPECTING VIRTUAL ADDRESS SPACE
05 12 099E 788 BNEQ 15$
50 1C B1 09A0 789 CMPW #SS$_EXQUOTA,R0 ;THEN EXCEEDS QUOTA MAY ALSO BE RETU
4B 13 09A3 790 BEQL 10$
09A5 791 15$:
53 01 B1 09A5 792 CMPW #SS$_NORMAL,R3 ;IF EXPECTING NORMAL COMPLETION
07 12 09A8 793 BNEQ 5$ ;FOR CRMPSC, THEN GLOBAL SECTION
50 0619 8F B1 09AA 794 CMPW #SS$_CREATED,R0 ;CREATED MAY BE RETURNED
3F 13 09AF 795 BEQL 10$
09B1 796
54 54 DD 09B1 797 5$: PUSHL R4
04 AE D0 09B3 798 MOVL 4(SP),R4 ;ADDRESS OF ERROR
09B7 799 $FAO_S (R1),MSGLEN,MSGBUFD,R4,R0,R3,-
09B7 800 INRANGE,INRANGE+4,RETRANGE,RETRANGE+4
10 BA 09EA 801 POPR #^M<R4>
0150 30 09EC 802 BSBW TYPEMSGBUF
05 09EF 803 RSB
09F0 804 10$:
007F 31 09F0 805 BRW RANGECHK ;GO CHECK THE RETURN RANGE
09F3 806 :
09F3 807 : INPUT:
09F3 808 :
09F3 809 : R1 = RETADR
09F3 810 : R3 = DESIRED STATUS
09F3 811 : R4 = PAGCNT
09F3 812 : R5 = REGION
09F3 813 :
09F3 814 : OUTPUT:
09F3 815 :
09F3 816 : R2 PRESERVED
09F3 817 :
09F3 818 CNTREGSUBR:
09F3 819 $CNTREG_S R4,(R1),R5
51 FCOA CF DE 0A02 820 MOVAL -W^CNTREGERR,R1 ;ERROR CONTROL STRING
14 11 0A07 821 BRB CHECK2
0A09 822 :
0A09 823 : INPUT:
0A09 824 :
0A09 825 : R1 = RETADR
0A09 826 : R3 = DESIRED STATUS
0A09 827 : R4 = PAGCNT
0A09 828 : R5 = REGION
0A09 829 :
0A09 830 : OUTPUT:
0A09 831 :
0A09 832 : R2 PRESERVED
0A09 833 :
0A09 834 EXPREGSUBR:
0A09 835 $EXPREG_S R4,(R1),R5
51 FC04 CF DE 0A18 836 MOVAL -W^EXPREGERR,R1 ;ERROR CONTROL STRING
0A1D 837 CHECK2:

```

-\$

Ps
--
\$C

MS

MS

MS

MS

MMGXQUOTA
Symbol table

- TEST MMG SERVICES WITH LOW PGFLQUOTA

K 15

16-SEP-1984 02:05:45 VAX/VMS Macro V04-00
5-SEP-1984 01:58:27 [MMGTST.SRC]MMGXQUOTA.MAR;1

Page 26
(10)

\$\$TAB	= 000000A0	R	02	FILENAME	000006A8	R	03
\$\$TABEND	= 000000E4	R	02	FILENAMEADR	000005F4	R	03
\$\$TMP	= 00000000			FILENAME\$IZ	= 0000000A		
\$\$TMP1	= 00000001			GBLSECNAM	000006A0	R	03
\$\$TMP2	= 000000CF			GBLSECNAMADR	000005EE	R	03
\$ST1	= 00000000			GBLSECNAM\$IZ	= 00000006		
\$ST2	= 00000004			GBL SEC MESSAGE	0000074C	R	03
ADJWSLERR	00000638	R	03	HIGHPOADR	00000018	R	02
ADJWSLERRADR	000002DD	R	03	IDMSG	00000678	R	03
ADJWSLERR\$IZ	= 00000068			IDMSGADR	000004ED	R	03
ADJWSLSUBR	00000AEC	R	03	IDMSG\$IZ	= 00000039		
BIT	= 00000004			INRANGE	00000000	R	02
CHECK1	00000994	R	03	L50	0000072A	R	03
CHECK2	00000A1D	R	03	LIB\$ CREMAPSEC	*****	X	03
CNTREGERR	00000610	R	03	LKWSETERR	00000628	R	03
CNTREGERRADR	000001F9	R	03	LKWSETERRADR	00000133	R	03
CNTREGERR\$IZ	= 00000072			LKWSETERR\$IZ	= 00000063		
CNTREGSUBR	000009F3	R	03	LKWSETSUBR	0000096E	R	03
CRETVAERR	00000600	R	03	MAIN PROGRAM	0000070B	R	03
CRETVAERRADR	0000000A	R	03	MAXPAGLOCK	0000089C	R	03
CRETVAERR\$IZ	= 00000063			MAXPASSCNT	00000020	R	02
CRETVASUBR	00000946	R	03	MAXWHICHRUN	00000028	R	02
CRETVA_MESSAGE	00000730	R	03	MEMLOOPCTL	00000668	R	03
CRLF	000000F8	R	02	MEMLOOPCTLADR	0000041F	R	03
CRMPSCERR	00000618	R	03	MEMLOOPCTL\$IZ	= 0000007F		
CRMPSCERRADR	000000D0	R	03	MSGBUF	00000106	R	02
CRMPSCERR\$IZ	= 00000063			MSGBUFD	000000E8	R	02
CTLSM_MEMLOOP	= 00000001			MSGBUFID	000000F8	R	02
CTLSM_PIDMSG	= 00000004			MSGBUF\$IZ	= 000000A0		
CTLSM_RNGCHK	= 00000008			MSGLEN	000000E4	R	02
CTLSM_TSTLOOP	= 00000002			NOREADERR	00000650	R	03
CTLSV_MEMLOOP	= 00000000			NOREADERRADR	000003B4	R	03
CTLSV_PIDMSG	= 00000002			NOREADERR\$IZ	= 00000024		
CTLSV_RNGCHK	= 00000003			NOWRITERR	00000660	R	03
CTLSV_TSTLOOP	= 00000001			NOWRITERRADR	000003FA	R	03
CTLFLG	00000010	R	02	NOWRITERR\$IZ	= 00000025		
DELTVAERR	00000608	R	03	OUTNAMADR	00000000	R	03
DELTVAERRADR	0000006D	R	03	OUTNAM\$IZ	= 0000000A		
DELTVAERR\$IZ	= 00000063			PASSCNT	00000024	R	02
DELTVASUBR	0000095A	R	03	PFNMAP_MESSAGE	0000076D	R	03
END	00000718	R	03	PID	0000001C	R	02
EXPREGERR	00000620	R	03	PIDCTL	00000698	R	03
EXPREGERRADR	00000209	R	03	PIDCTLADR	000005EB	R	03
EXPREGERR\$IZ	= 00000072			PIDCTL\$IZ	= 00000003		
EXPREGSUBR	00000A09	R	03	PIDMSG	00000102	R	02
FAB	00000050	R	02	PIDMSGD	000000F0	R	02
FAB\$C_BID	= 00000003			PRESTART	00000724	R	03
FAB\$C_BLN	= 00000050			PREVPROT	00000050	R	02
FAB\$C_SEQ	= 00000000			PROBERR	00000879	R	03
FAB\$C_VAR	= 00000002			PRT\$C_NONE	= 00000010		
FAB\$S_ALQ	= 00000010			RAB	000000A0	R	02
FAB\$S_FOP	= 00000004			RAB\$B_RAC	= 0000001E		
FAB\$V_CHAN_MODE	= 00000002			RAB\$C_BID	= 00000001		
FAB\$V_FILE_MODE	= 00000004			RAB\$C_BLN	= 00000044		
FAB\$V_LNM_MODE	= 00000000			RAB\$C_SEQ	= 00000000		
FAB\$V_PUT	= 00000000			RAB\$S_CTX	= 00000018		
FAB\$W_GBC	= 0000C.48			RAB\$S_RBF	= 00000028		

! 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
DATA0	000001A6 (422.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC PAGE
CODE	00000BDF (3039.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC PAGE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	18	00:00:00.06	00:00:01.31
Command processing	89	00:00:00.80	00:00:05.45
Pass 1	370	00:00:13.25	00:00:42.30
Symbol table sort	0	00:00:01.28	00:00:04.04
Pass 2	207	00:00:03.49	00:00:12.96
Symbol table output	22	00:00:00.17	00:00:01.22
Psect synopsis output	5	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	713	00:00:19.08	00:01:07.31

The working set limit was 1650 pages.
78987 bytes (155 pages) of virtual memory were used to buffer the intermediate code.
There were 50 pages of symbol table space allocated to hold 931 non-local and 30 local symbols.
980 source lines were read in Pass 1, producing 23 object records in Pass 2.
63 pages of virtual memory were used to define 52 macros.

! Macro library statistics !

Macro library name	Macros defined
_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	1
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	32
TOTALS (all libraries)	33

1185 GETS were required to define 33 macros.

There were no errors, warnings or information messages.

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

