



```

MM    MM    MM    MM    GGGGGGGG    SSSSSSSS    EEEEEEEEEE    TTTTTTTTTT    PPPPPPPP    RRRRRRRR    TTTTTTTTTT
MM    MM    MM    MM    GGGGGGGG    SSSSSSSS    EEEEEEEEEE    TTTTTTTTTT    PPPPPPPP    RRRRRRRR    TTTTTTTTTT
MMMM  MMMM  MMMM  MMMM  GG          SS          EE          TT          PP          PP  RR          RR    TT
MMMM  MMMM  MMMM  MMMM  GG          SS          EE          TT          PP          PP  RR          RR    TT
MM  MM  MM  MM  MM  MM  GG          SS          EE          TT          PP          PP  RR          RR    TT
MM  MM  MM  MM  MM  MM  GG          SS          EE          TT          PP          PP  RR          RR    TT
MM    MM    MM    MM    GG          SS          EE          TT          PPPPPPPP  RRRRRRRR    TT
MM    MM    MM    MM    GG          SS          EE          TT          PPPPPPPP  RRRRRRRR    TT
MM    MM    MM    MM    GG    GGGGGG        SS          EE          TT          PP          RR  RR      TT
MM    MM    MM    MM    GG    GGGGGG        SS          EE          TT          PP          RR  RR      TT
MM    MM    MM    MM    GG          GG          SS          EE          TT          PP          RR  RR      TT
MM    MM    MM    MM    GG          GG          SS          EE          TT          PP          RR  RR      TT
MM    MM    MM    MM    GGGGGG        SSSSSSSS    EEEEEEEEEE    TT          PP          RR          RR    TT
MM    MM    MM    MM    GGGGGG        SSSSSSSS    EEEEEEEEEE    TT          PP          RR          RR    TT

```

```

LL                IIIIIII  SSSSSSSS
LL                IIIIIII  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
LLLLLLLLLLLLL   IIIIIII  SSSSSSSS
LLLLLLLLLLLLL   IIIIIII  SSSSSSSS

```

(2)	48	DECLARATIONS
(2)	52	MACROS
(3)	199	DATA STORAGE AND MESSAGE STRINGS
(6)	326	INITIALIZATION
(7)	374	FORCE ERRORS FROM SETPRT
(8)	411	SUBROUTINES TO CALL THE SERVICES
(9)	554	MISCELLANEOUS SUBROUTINES

```

0000 1 :
0000 2 :
0000 3 :
0000 4 :
0000 5 :
0000 6 :
0000 7 :
0000 8 :
0000 9 :
0000 10 :
0000 11 :
0000 12 :
0000 13 :
0000 14 :
0000 15 :
0000 16 :
0000 17 :
0000 18 :
0000 19 :
0000 20 :
0000 21 :
0000 22 :
0000 23 :
0000 24 :
0000 25 :
0000 26 :
0000 27 :
0000 28 :
0000 29 :
0000 30 :
0000 31 :
0000 32 :
0000 33 :
0000 34 :
0000 35 :
0000 36 :
0000 37 :
0000 38 :
0000 39 :
0000 40 :
0000 41 :
0000 42 :
0000 43 :
0000 44 :
0000 45 :
0000 46 :

```

MEMORY MANAGEMENT SERVICES TEST #2

```

.TITLE MMGSETPRT - TEST OF $SETPRT SYSTEM SERVICE
.IDENT 'V04-000'
*****
*
*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
*   ALL RIGHTS RESERVED.
*
*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
*   ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
*   OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
*   TRANSFERRED.
*
*   THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
*   AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
*   CORPORATION.
*
*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****
*
*++
FACILITY: USER MODE MEMORY MANAGEMENT SERVICES TEST
ABSTRACT: THIS SET OF ROUTINES TESTS THE MEMORY MANAGEMENT SERVICES
ENVIRONMENT: USER MODE DIAGNOSTIC
AUTHOR: PETER H. LIPMAN      , CREATION DATE: 6-JAN-77
MODIFIED BY:
          V02-012 SHZ0007      Stephen Zalewski      20-Aug-1980
          Added further tests to system services tested in this
          program. Also incorporated program into MMG test
          package.

```

```
0000 48      .SBTTL  DECLARATIONS
0000 49      :
0000 50      : INCLUDE FILES:
0000 51      :
0000 52      :      .SBTTL  MACROS
0000 53      :
0000 54      :      MACROS:
0000 55      :
0000 56      .MACRO  LIST
0000 57      .LIST  MEB
0000 58      .ENDM  LIST
0000 59
0000 60      .MACRO  NLIST
0000 61      .NLIST  MEB
0000 62      .ENDM  NLIST
0000 63
0000 64      .MACRO  READ  SIZ=#1,ADR=(R2),?L1,?L2
0000 65      IFNORD  <SIZ>,<ADR>,L1
0000 66      BRB     L2
0000 67 L1:    MOVAL  W^READERR,R1
0000 68      BSBW   PROBERR
0000 69 L2:
0000 70      .ENDM  READ
0000 71
0000 72      .MACRO  WRITE SIZ=#1,ADR=(R2),?L1,?L2
0000 73      IFNOWRT <SIZ>,<ADR>,L1
0000 74      BRB     L2
0000 75 L1:    MOVAL  W^WRITERR,R1
0000 76      BSBW   PROBERR
0000 77 L2:
0000 78      .ENDM  WRITE
0000 79
0000 80      .MACRO  NOREAD SIZ=#1,ADR=(R2),?L1
0000 81      IFNORD  <SIZ>,<ADR>,L1
0000 82      MOVAL  W^NOREADERR,R1
0000 83      BSBW   PROBERR
0000 84 L1:
0000 85      .ENDM  NOREAD
0000 86
0000 87      .MACRO  NOWRITE SIZ=#1,ADR=(R2),?L1
0000 88      IFNOWRT <SIZ>,<ADR>,L1
0000 89      MOVAL  W^NOWRITERR,R1
0000 90      BSBW   PROBERR
0000 91 L1:
0000 92      .ENDM  NOWRITE
0000 93
0000 94      .MACRO  CRETVA STARTVA,ENDVA,STATUS=S^#SS$ NORMAL,-
0000 95      LIST      INADR=W^INRANGE,RETADR=W^RETRANGE
0000 96
0000 97      .IF     NB,STARTVA
0000 98      MOVL   STARTVA,W^INRANGE
0000 99      .ENDC
0000 100     .IF     NB,ENDVA
0000 101     MOVL   ENDVA,W^INRANGE+4
0000 102     .ENDC
0000 103     MOVZWL STATUS,R3
0000 104     MOVAL  INADR,R0
```

```

0000 105          MOVAL  RETADR,R1
0000 106          BSBW   CRETVA$UBR
0000 107          NLIST
0000 108          .ENDM  CRETVA
0000 109
0000 110          .MACRO DELTVA STARTVA,ENDVA,STATUS=S^#SS$ NORMAL,-
0000 111                    INADR=W^INRANGE,RETADR=W^RETRANGE
0000 112          LIST
0000 113                    .IF    NB,STARTVA
0000 114                    MOVL  STARTVA,W^INRANGE
0000 115                    .ENDC
0000 116                    .IF    NB,ENDVA
0000 117                    MOVL  ENDVA,W^INRANGE+4
0000 118                    .ENDC
0000 119                    MOVZWL STATUS,R3
0000 120                    MOVAL  INADR,R0
0000 121                    MOVAL  RETADR,R1
0000 122                    BSBW   DELTVA$UBR
0000 123          NLIST
0000 124          .ENDM  DELTVA
0000 125
0000 126          .MACRO EXPREG  PAGCNT,REGION=#0,STATUS=S^#SS$_NORMAL,-
0000 127                    RETADR=W^RETRANGE
0000 128          LIST
0000 129                    MOVZWL STATUS,R3
0000 130                    MOVL  PAGCNT,R4
0000 131                    MOVAL  RETADR,R1
0000 132                    .IF    IDN,<REGION>,<#0>
0000 133                    CLRL  R5
0000 134                    .IFF
0000 135                    MOVL  REGION,R5
0000 136                    .ENDC
0000 137                    BSBW   EXPREG$SUBR
0000 138          NLIST
0000 139          .ENDM  EXPREG
0000 140
0000 141          .MACRO SETPRT ACC,STARTVA,ENDVA,STATUS=S^#SS$ NORMAL,-
0000 142                    INADR=W^INRANGE,RETADR=W^RETRANGE,-
0000 143                    PRVPRT=NONE,PRVPRTADR
0000 144          LIST
0000 145                    .IF    NB,STARTVA
0000 146                    MOVL  STARTVA,W^INRANGE
0000 147                    .ENDC
0000 148                    .IF    NB,ENDVA
0000 149                    MOVL  ENDVA,W^INRANGE+4
0000 150                    .ENDC
0000 151                    MOVZWL STATUS,R3
0000 152                    MOVAL  INADR,R0
0000 153                    MOVAL  RETADR,R1
0000 154                    MOVZBL S^#PRT$C-'ACC',R4
0000 155                    MOVZBL S^#PRT$C-'PRVPRT',R5
0000 156                    .IF    B,PRVPRTADR
0000 157                    .IF    DIF,<PRVPRT>,<NONE>
0000 158                    MOVAL  W^PREVPROT,R6
0000 159                    .IFF
0000 160                    CLRL  R6
0000 161                    .ENDC

```

```

0000 162          .IFF
0000 163          MOVAL  PRVPRTADR,R6
0000 164          .ENDC
0000 165          BSBW   SETPRTSUBR
0000 166          NLIST
0000 167          .ENDM  SETPRT
0000 168
0000 169          .MACRO RANGECHK ONOROFF
0000 170          LIST
0000 171          .IF    IDN <ONOROFF>,<OFF>
0000 172          BICL   #CTLSM_RNGCHK,W^CTLFLG
0000 173          .IFF
0000 174          BISL   #CTLSM_RNGCHK,W^CTLFLG
0000 175          .ENDC
0000 176          NLIST
0000 177          .ENDM  RANGECHK
0000 178
0000 179          :
0000 180          : EQUATED SYMBOLS:
0000 181          :
0000 182          $SECDEF
0000 183          $SSDEF
0000 184          $PRTDEF
0000 185          $GBLINI
0000 186          $VIELD CTL,0,<-
0000 187          <MEMLOOP,,MASK>,-
0000 188          <TSTLOOP,,MASK>,-
0000 189          <PIDMSG,,MASK>,-
0000 190          <RNGCHK,,MASK>-
0000 191          >
00000010 0000 192          PRT$C_NONE=1a4
00000020 0000 193          PRT$C_TOBIG=1a5
00000001 0000 194          PRT$C_RESERVE=1a0
0000 195          :
0000 196          : OWN STORAGE:
0000 197          :

```

```

: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 199 .SBTTL DATA STORAGE AND MESSAGE STRINGS
00000000 200 .PSECT DATA0,PAGE,WRT,NOEXE
0000 201 INRANGE:
00000008 0000 202 .BLKL 2
0008 203 RETRANGE:
00000010 0008 204 .BLKL 2
00000006 0010 205 CTLFLG: .LONG CTL$M_TSTLOOP!CTL$M_PIDMSG
00000018 0014 206 SAVEND: .BLKL 1
0000001C 0018 207 PID: .BLKL 1 ;PROCESS ID
001C 208 MAXPASSCNT:
00000003 001C 209 .LONG 3 ;NUMBER OF PASSES TO RUN
0020 210 PASSCNT:
00000024 0020 211 .BLKL 1 ;PASS COUNTER
0024 212 PREVPROT:
0024 213 FAB: $FAB FAC-PUT, FNA=OUTNAMADR, FNS=OUTNAMSIZ ;FAB FOR OUTPUT
0074 214 RAB: $RAB FAB=FAB ;RECORD ACCESS BLOCK FOR OUTPUT
000000BC 00B8 215 MSGLEN: .BLKL 1 ;RETURN LENGTH FROM FAO
000000DA'000000A0' 00BC 216 MSGBUFD: .LONG MSGBUFSIZ,MSGBUF ;MESSAGE BUFFER DESCRIPTOR
00C4 217 PIDMSGD:
000000D6'00000004' 00C4 218 .LONG MSGBUF-PIDMSG,PIDMSG
00CC 219 :
00CC 220 : ***** DO NOT SEPARATE OR REORDER THE FOLLOWING LINES
00CC 221 :
00CC 222 MSGBUFID:
00CC 223 CRLF: .BYTE ^015,^012
20 53 53 45 43 4F 52 50 00CE 224 .ASCII $PROCESS $
20 20 20 20 00D6 225 PIDMSG: .ASCII $ $
0000017A 00DA 226 MSGBUF: .BLKB 160 ;MESSAGE BUFFER USED BY FAO
000000A0 017A 227 MSGBUFSIZ=-MSGBUF
017A 228 :
017A 229 : ***** DO NOT SEPARATE OR REORDER THE PRECEEDING LINES
017A 230 :
017A 231 :

```

MM  
Sy  
SS  
SS  
SS  
SS  
SS  
SS  
BI  
CH  
CH  
CR  
CR  
CR  
CR  
CR  
CT  
CT  
CT  
CT  
CT  
CT  
CT  
DE  
DE  
DE  
DE  
EX  
EX  
EX  
EX  
FA  
FA  
FA  
FA  
FA  
FA  
FA  
FA  
FA  
FA  
ID  
ID  
ID  
IN  
MA  
MS  
MS  
MS  
MS  
NO  
NO  
NO



```

00000000 233
0000 234
0000 235
54 55 50 54 55 4F 24 53 59 53 0000 236
0000000A 000A 237
000A 238
000A 239
52 52 45 20 41 56 54 45 52 43 2F 21 000A 240
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 241
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 006D 242
006D 243
006D 244
52 52 45 20 41 56 54 4C 45 44 2F 21 006D 245
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 246
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 00D0 247
00D0 248
00D0 249
52 52 45 20 47 45 52 50 58 45 2F 21 00D0 250
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
20 3D 20 54 4E 43 47 41 50 09 2F 21 0108 251
20 4E 4F 49 47 45 52 20 2C 4C 53 21 0114
45 43 41 50 53 20 42 55 21 50 20 3D 0120
012C
4C 58 21 20 3D 20 52 44 41 54 45 52 012E 252
013A
00000072 0142 253
0142 254
0142 255
52 52 45 20 54 52 50 54 45 53 2F 21 0142 256
58 21 20 3D 20 43 50 20 2D 20 52 4F 014E
41 57 20 53 55 54 41 54 53 20 2C 4C 015A
4C 55 4F 48 53 20 2C 4C 58 21 20 53 0166
0172
21 20 3D 20 52 44 41 4E 49 09 2F 21 017A 257
52 20 20 2C 4C 58 21 20 2D 20 4C 58 0186
20 4C 58 21 20 3D 20 52 44 41 54 45 0192
019E
20 53 55 4F 49 56 45 52 50 09 2F 21 01A3 258
57 20 4E 4F 49 54 43 45 54 4F 52 50 01AF
4F 48 53 20 2C 42 58 31 21 20 53 41 01BB

```

```

.PSECT CODE,PAGE,NOWRT,EXE
OUTNAMADR:
.ASCII /SYSS$OUTPUT/
OUTNAMSIZ=-OUTNAMADR
CRETVAERRADR:
.ASCII $!/CRETVA ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
241
.ASCII $!/ INADR = !XL - !XL, RETADR = !XL - !XL!/$
CRETVAERRSIZ=-CRETVAERRADR
242
DELTVAERRADR:
.ASCII $!/DELTVA ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
243
244
245
.ASCII $!/ INADR = !XL - !XL, RETADR = !XL - !XL!/$
DELTVAERRSIZ=-DELTVAERRADR
246
247
248
EXPREGERRADR:
.ASCII $!/EXPREG ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
249
250
.ASCII $!/ PAGCNT = !SL, REGION = P!UB SPACE, $
251
.ASCII $RETADR = !XL - !XL!/$
EXPREGERRSIZ=-EXPREGERRADR
252
253
254
255
SETPRTERRADR:
.ASCII $!/SETPRT ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
256
257
.ASCII $!/ INADR = !XL - !XL, RETADR = !XL - !XL$
258
.ASCII $!/ PREVIOUS PROTECTION WAS !1XB, SHOULD BE !1XB!/$

```

MM  
SY  
SY  
SY  
SY  
SY  
SY  
SY  
SY  
TY  
WR  
WR  
  
PS  
--  
\$A  
DA  
CO  
  
Ph  
--  
In  
Co  
Pa  
Sy  
Pa  
Sy  
Ps  
Cr  
As  
  
Th  
70  
Th  
59  
50  
  
Ma  
--  
\$  
\$  
TO

21 42 58 31 21 20 45 42 20 44 4C 55	01C7		
	2F 01D3		
	00000092	259	SETPRTERRSIZ=-SETPRTERRADR
	01D4	260	
	01D4	261	READERRADR:
4F 52 52 45 20 44 41 45 52 20 2F 21	01D4	262	.ASCII \$!/ READ ERROR - LOCATION = !XL !/\$
4E 4F 49 54 41 43 4F 4C 20 2D 20 52	01E0		
	2F 21 20 4C 58 21 20 3D 20		
	00000021	263	READERRSIZ=-READERRADR
	01F5	264	
	01F5	265	NOREADERRADR:
45 20 44 41 45 52 2D 4F 4E 20 2F 21	01F5	266	.ASCII \$!/ NO-READ ERROR - LOCATION = !XL !/\$
54 41 43 4F 4C 20 2D 20 52 4F 52 52	0201		
2F 21 20 4C 58 21 20 3D 20	020D		
	00000024	267	NOREADERRSIZ=-NOREADERRADR
	0219	268	
	0219	269	WRITERRADR:
52 52 45 20 45 54 49 52 57 20 2F 21	0219	270	.ASCII \$!/ WRITE ERROR - LOCATION = !XL !/\$
4F 49 54 41 43 4F 4C 20 2D 20 52 4F	0225		
	2F 21 20 4C 58 21 20 3D 20 4E		
	00000022	271	WRITERRSIZ=-WRITERRADR
	023B	272	
	023B	273	NOWRITERRADR:
20 45 54 49 52 57 2D 4F 4E 20 2F 21	023B	274	.ASCII \$!/ NO-WRITE ERROR - LOCATION = !XL !/\$
41 43 4F 4C 20 2D 20 52 4F 52 52 45	0247		
21 20 4C 58 21 20 3D 20 4E 4F 49 54	0253		
	2F 025F		
	00000025	275	NOWRITERRSIZ=-NOWRITERRADR
	0260	276	
	0260	277	RANGERRADR:
4E 41 52 20 4E 52 55 54 45 52 2F 21	0260	278	.ASCII \$!/RETURN RANGE ERROR - LOCATION = !XLS
4C 20 2D 20 52 4F 52 52 45 20 45 47	026C		
58 21 20 3D 20 4E 4F 49 54 41 43 4F	0278		
	4C 0284		
21 20 3D 20 52 44 41 4E 49 09 2F 21	0285	279	.ASCII \$!/ INADR = !XL - !XL, RETADR = !XL - !XL!/\$
45 52 20 2C 4C 58 21 20 2D 20 4C 58	0291		
2D 20 4C 58 21 20 3D 20 52 44 41 54	029D		
	2F 21 4C 58 21 20		
	0000004F	280	RANGERRSIZ=-RANGERRADR
	02AF	281	
	02AF	282	IDMSGADR:
4E 41 4D 20 59 52 4F 4D 45 4D 2F 21	02AF	283	.ASCII \$!/MEMORY MANAGEMENT SERVICES TEST #2 (SETPRT), PASS !UL!/\$
56 52 45 53 20 54 4E 45 4D 45 47 41	02BB		
32 23 20 54 53 45 54 20 53 45 43 49	02C7		
50 20 2C 29 54 52 50 54 45 53 28 20	02D3		
	2F 21 4C 55 21 20 53 53 41		
	00000039	284	IDMSGISIZ=-IDMSGADR
	02E8	285	
	02E8	286	RUN1_MSGADR:
20 20 2A 2A 2A 2A 2A 20 20 20 2F 21	02E8	287	.ASCII \$!/ ***** TEST WILL NOW BE RUN USING NORMAL VA SPACE *****\$
4E 20 4C 4C 49 57 20 54 53 45 54 20	02F4		
53 55 20 4E 55 52 20 45 42 20 57 4F	0300		
56 20 4C 41 4D 52 4F 4E 20 47 4E 49	030C		
2A 2A 20 20 20 45 43 41 50	0318		
	2A 2A 2A		
	20 2F 21	288	.ASCII \$!/ \$
	00000042	289	RUN1_MSGSIZ=-RUN1_MSGADR
	032A		

MMG  
VA)  
114  
The  
MAC

MMGSETPRT  
V04-000

- TEST OF S\$ETPRT SYSTEM SERVICE J 12  
DATA STORAGE AND MESSAGE STRINGS

16-SEP-1984 01:59:27  
5-SEP-1984 01:58:23

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

Page 8  
(4)

	032A	290	
	032A	291	PIDCTLADR:
4C 55 21	032A	292	.ASCII \$!UL\$
00000003	032D	293	PIDCTLSIZ=-PIDCTLADR

\*\*

```

032D 295 :
032D 296 : STRING DESCRIPTORS
032D 297 :
032D 298 .ALIGN LONG
0330 299
0330 300 CRETVAERR:
0000000A'00000063 0330 301 .LONG CRETVAERRSIZ,CRETVAERRADR
0338 302 DELTVAERR:
0000006D'00000063 0338 303 .LONG DELTVAERRSIZ,DELTVAERRADR
0340 304 EXPREGERR:
000000D0'00000072 0340 305 .LONG EXPREGERRSIZ,EXPREGERRADR
0348 306 SETPRTERR:
00000142'00000092 0348 307 .LONG SETPRTERRSIZ,SETPRTERRADR
0350 308 READERR:
000001D4'00000021 0350 309 .LONG READERRSIZ,READERRADR
0358 310 NOREADERR:
000001F5'00000024 0358 311 .LONG NOREADERRSIZ,NOREADERRADR
0360 312 WRITERR:
00000219'00000022 0360 313 .LONG WRITERRSIZ,WRITERRADR
0368 314 NOWRITERR:
0000023B'00000025 0368 315 .LONG NOWRITERRSIZ,NOWRITERRADR
0370 316 RANGERR:
00000260'0000004F 0370 317 .LONG RANGERRSIZ,RANGERRADR
0378 318 IDMSG:
000002AF'00000039 0378 319 .LONG IDMSGSIZ,IDMSGADR
0380 320 RUN1_MSG:
000002E8'00000042 0380 321 .LONG RUN1_MSGSIZ,RUN1_MSGADR
0388 322 PIDCTL:
0000032A'00000003 0388 323 .LONG PIDCTLSIZ,PIDCTLADR
0390 324

```

```

0390 326 .SBTTL INITIALIZATION
0390 327 :*****
0390 328 :PROGRAM DESCRIPTION:
0390 329 :
0390 330 :   THIS PROGRAM TESTS THE FOLLOWING SYSTEM SERVICE:
0390 331 :   $SETPRT
0390 332 :
0390 333 :   THE PROGRAM FORCES POSSIBLE ERROR PATHS FOR THE ABOVE MENTIONED
0390 334 :   SYSTEM SERVICES.  THREE PASSES ARE MADE THROUGH THE TEST LOOP
0390 335 :   TO ENSURE PATH REPEATABILITY.  ONLY REGULAR VA SPACE IS USED IN
0390 336 :   THIS TEST PROGRAM.
0390 337 :
0390 338 :   REFER TO MASDS:[MMGSTS.COM]MMGTST.RAP FOR FURTHER INFORMATION
0390 339 :   REGARDING JUST HOW COMPLETELY THE ABOVE MENTIONED SYSTEM SERVICES
0390 340 :   ARE TESTED BY THIS PROGRAM.
0390 341 :
0390 342 :   *PRIVILEGES:
0390 343 :   THIS PROGRAM NEEDS NO SPECIAL PRIVILEGES TO EXECUTE.
0390 344 :*****
0390 345 :
0390 346 : START HERE
0390 347 :
0000 0390 348 START: .WORD 0 ;ENTRY MASK
OE 50 E9 0392 349 $OPEN W^FAB ;OPEN THE FILE '$OUTPUT'
0390 350 BLBC RO,10$ ;BRANCH IF ERROR
09 50 E8 03A0 351 $CONNECT W^RAB ;CONNECT THE RECORD ACCESS BLOCK
0390 352 BLBS RO,20$
0390 353 10$: $EXIT_S RO ;EXIT WITH STATUS IN RO
00000020'EF 01 D0 03B7 354 20$: MOVL #1,PASSCNT ;INITIALIZE THE PASS COUNT
50 00000018'EF 3C 03BE 355 $RESUME_S PID ;SET UP PROCESS ID
0390 356 MOVZWL PID,RO
0390 357 $FAO_S PIDCTL,MSGLEN,PIDMSGD,RO ;INIT THE PROCESS ID STRING
0390 358 :
0390 359 : INFORM OPERATOR THAT TESTS WILL BE RUN USING ONLY NORMAL VA SPACE
0390 360 :
0390 361 $FAO_S RUN1 MSG,MSGLEN,MSGBUFD ;INFORM OPR NORMAL VA USED FOR TEST
0010'CF 03FF 30 0402 362 BSBW- TYPEMSGBUF
0405 363 BICL #CTL$M_PIDMSG,W^CTLFLG ;STOP PROCESS ID FROM PRINTING
040A 364 RSTART: RANGECHK ON
040A 365 BICL #CTL$M_RNGCHK,W^CTLFLG
0390 366 $FAO_S IDMSG,MSGLEN,MSGBUFD,PASSCNT
0390 367 BSBW- TYPEMSGBUF
0390 368 EXPREG #1
53 01 3C 042F MOVZWL S^#SS$_NORMAL,R3
54 01 D0 0432 MOVL #1,R4
51 0008'CF DE 0435 MOVAL W^RETRANGE,R1
55 D4 043A CLRL R5
0272 30 043C BSBW EXPREGSUBR
52 0008'CF 7D 043F 369 MOVQ W^RETRANGE,R2
0000'CF 52 7D 0444 370 MOVQ R2,W^INRANGE
0014'CF 52 D0 0449 371 MOVL R2,W^SAVEND
044E 372

```

```

044E 374 .SBTTL FORCE ERRORS FROM SETPRT
044E 375 :
044E 376 : FORCE ERRORS IN SETPRT
044E 377 :
044E 378 :
0010'CF 08 CA 044E RANGECHK OFF
                    BICL #CTLSM_RNGCHK,W^CTLFLG
                    DELTVA #CTLSM_RNGCHK,W^CTLFLG ;DELETE THE JUNK
                    53 01 3C 0453
50 0000'CF DE 0453 MOVZWL S^#SS$ NORMAL,R3
51 0008'CF DE 0456 MOVZWL S^#SS$ NORMAL,R3
                    01E7 30 045B MOVAL W^INRANGE,R0
                    01E7 30 0460 MOVAL W^RETRANGE,R1
                    0463 380 BSBW DELTVASUBR
0010'CF 08 C8 0463 RANGECHK ON
                    BISL #CTLSM_RNGCHK,W^CTLFLG
                    53 01 3C 0468 381 SETPRT UW,STATUS=#SS$ LENVIO ;PROTECT OFF END OF PO SPACE
                    50 0000'CF DE 0468 MOVZWL #SS$ LENVIO,R3
                    51 0008'CF DE 046D MOVAL W^INRANGE,R0
                    54 04 9A 0472 MOVAL W^RETRANGE,R1
                    55 10 9A 0477 MOVZBL S^#PRT$C_UW,R4
                    56 10 9A 047A MOVZBL S^#PRT$C_NONE,R5
                    0298 30 047D CLRL R6
                    0298 30 047F BSBW SETPRTSUBR
0000'CF 04 DO 0482 382 SETPRT UW,#4,#8,#SS$ ACCVIO ;DELETED PAGE
0004'CF 08 DO 0482 MOVL #4,W^INRANGE
                    53 0C 3C 0487 MOVL #8,W^INRANGE+4
50 0000'CF DE 048C MOVZWL #SS$ ACCVIO,R3
51 0008'CF DE 048F MOVAL W^INRANGE,R0
                    54 04 9A 0494 MOVAL W^RETRANGE,R1
                    55 10 9A 0499 MOVZBL S^#PRT$C_UW,R4
                    56 10 9A 049C MOVZBL S^#PRT$C_NONE,R5
                    0276 30 049F CLRL R6
                    0276 30 04A1 BSBW SETPRTSUBR
0000'CF 80000200 8F DO 04A4 383 SETPRT UR,#^X80000200,#^X80000A00,#SS$ NOPRIV
0004'CF 80000A00 8F DO 04A4 MOVL #^X80000200,W^INRANGE
                    53 24 3C 04AD MOVL #^X80000A00,W^INRANGE+4
50 0000'CF DE 04B6 MOVZWL #SS$ NOPRIV,R3
51 0008'CF DE 04B9 MOVAL W^INRANGE,R0
                    54 0F 9A 04BE MOVAL W^RETRANGE,R1
                    55 10 9A 04C3 MOVZBL S^#PRT$C_UR,R4
                    56 10 9A 04C6 MOVZBL S^#PRT$C_NONE,R5
                    024C 30 04C9 CLRL R6
                    024C 30 04CB BSBW SETPRTSUBR
0000'CF 7FFEFFFF 8F DO 04CE 384 SETPRT UR,<#1@31-<128@9>-1>,W^INRANGE,#SS$ PAGOWNVIO
0004'CF 0000'CF DO 04CE MOVL #1@31-<128@9>-1,W^INRANGE
                    53 01EC 8F 3C 04D7 MOVL W^INRANGE,W^INRANGE+4
50 0000'CF DE 04DE MOVZWL #SS$ PAGOWNVIO,R3
51 0008'CF DE 04E3 MOVAL W^INRANGE,R0
                    54 0F 9A 04E8 MOVAL W^RETRANGE,R1
                    55 10 9A 04ED MOVZBL S^#PRT$C_UR,R4
                    56 10 9A 04F0 MOVZBL S^#PRT$C_NONE,R5
                    0222 30 04F3 CLRL R6
0014'CF 00000600 8F C1 04F5 385 ADDL3 #^X600,W^SAVEND,W^INRANGE+4
0004'CF 0004'CF DO 04F8 386 CRETVA W^SAVEND,W^INRANGE+4
                    53 01 3C 0504 MOVL W^SAVEND,W^INRANGE
                    0004'CF 0004'CF DO 050B MOVL W^INRANGE+4,W^INRANGE+4
                    53 01 3C 0512 MOVZWL S^#SS$ NORMAL,R3

```

50	0000'CF	DE	0515		MOVAL	W^INRANGE,R0	
51	0008'CF	DE	051A		MOVAL	W^RETRANGE,R1	
	0114	30	051F		BSBW	CRETVASUBR	
			0522	387	SETPRT	URSW,STATUS=#SS\$ _ACCVIO,-	
			0522	388		INADR=W^4	;INPUT RANGE NOT ACCESSIBLE
	53 0C	3C	0522		MOVZWL	#SS\$ ACCVIO,R3	
50	0004'CF	DE	0525		MOVAL	W^4,R0	
51	0008'CF	DE	052A		MOVAL	W^RETRANGE,R1	
	54 0C	9A	052F		MOVZBL	S^#PRT\$C_URSW,R4	
	55 10	9A	0532		MOVZBL	S^#PRT\$C_NONE,R5	
	56	D4	0535		CLRL	R6	
	01E0	30	0537		BSBW	SETPRTSUBR	
			053A	389	SETPRT	UW	
	53 01	3C	053A		MOVZWL	S^#SS\$ NORMAL,R3	
50	0000'CF	DE	053D		MOVAL	W^INRANGE,R0	
51	0008'CF	DE	0542		MOVAL	W^RETRANGE,R1	
	54 04	9A	0547		MOVZBL	S^#PRT\$C_UW,R4	
	55 10	9A	054A		MOVZBL	S^#PRT\$C_NONE,R5	
	56	D4	054D		CLRL	R6	
	01C8	30	054F		BSBW	SETPRTSUBR	
			0552	390	SETPRT	URSW,STATUS=#SS\$ _ACCVIO,-	
			0552	391		RETADR=W^8	;RETURN RANGE NOT ACCESSIBLE
	53 0C	3C	0552		MOVZWL	#SS\$ ACCVIO,R3	
50	0000'CF	DE	0555		MOVAL	W^INRANGE,R0	
51	0008'CF	DE	055A		MOVAL	W^8,R1	
	54 0C	9A	055F		MOVZBL	S^#PRT\$C_URSW,R4	
	55 10	9A	0562		MOVZBL	S^#PRT\$C_NONE,R5	
	56	D4	0565		CLRL	R6	
	01B0	30	0567		BSBW	SETPRTSUBR	
			056A	392	SETPRT	UW	
	53 01	3C	056A		MOVZWL	S^#SS\$ NORMAL,R3	
50	0000'CF	DE	056D		MOVAL	W^INRANGE,R0	
51	0008'CF	DE	0572		MOVAL	W^RETRANGE,R1	
	54 04	9A	0577		MOVZBL	S^#PRT\$C_UW,R4	
	55 10	9A	057A		MOVZBL	S^#PRT\$C_NONE,R5	
	56	D4	057D		CLRL	R6	
	0198	30	057F		BSBW	SETPRTSUBR	
			0582	393	SETPRT	URSW,STATUS=#SS\$ _ACCVIO,-	
			0582	394		RETADR=@W^INRANGE	;CHANGE PROTECTION OF RETURN RANGE P
	53 0C	3C	0582		MOVZWL	#SS\$ ACCVIO,R3	
50	0000'CF	DE	0585		MOVAL	W^INRANGE,R0	
51	0000'DF	DE	058A		MOVAL	@W^INRANGE,R1	
	54 0C	9A	058F		MOVZBL	S^#PRT\$C_URSW,R4	
	55 10	9A	0592		MOVZBL	S^#PRT\$C_NONE,R5	
	56	D4	0595		CLRL	R6	
	0180	30	0597		BSBW	SETPRTSUBR	
			059A	395	SETPRT	UW	
	53 01	3C	059A		MOVZWL	S^#SS\$ NORMAL,R3	
50	0000'CF	DE	059D		MOVAL	W^INRANGE,R0	
51	0008'CF	DE	05A2		MOVAL	W^RETRANGE,R1	
	54 04	9A	05A7		MOVZBL	S^#PRT\$C_UW,R4	
	55 10	9A	05AA		MOVZBL	S^#PRT\$C_NONE,R5	
	56	D4	05AD		CLRL	R6	
	0168	30	05AF		BSBW	SETPRTSUBR	
			05B2	396	SETPRT	URSW,STATUS=#SS\$ _ACCVIO,-	
			05B2	397		PRVPRTADR=@W^INRANGE	;CHANGE PROTECTION OF PREVIOUS PROT
	53 0C	3C	05B2		MOVZWL	#SS\$ _ACCVIO,R3	

```

50 0000'CF DE 05B5 MOVAL W^INRANGE,R0
51 0008'CF DE 05BA MOVAL W^RETRANGE,R1
54 54 9A 05BF MOVZBL S^#PRT$C_URSW,R4
55 55 J 9A 05C2 MOVZBL S^#PRT$C_NONE,R5
56 0000'DF DE 05C5 MOVAL @W^INRANGE,R6
014D 30 05CA BSBW SETPRTSUBR
398 SETPRT TOBIG,STATUS=#SS$ IVPROTECT ;PROTECTION CODE GREATER THAN 15
53 02F4 8F 3C 05CD MOVZWL #SS$ IVPROTECT,R3
50 0000'CF DE 05D2 MOVAL W^INRANGE,R0
51 0008'CF DE 05D7 MOVAL W^RETRANGE,R1
54 54 20 9A 05DC MOVZBL S^#PRT$C_TOBIG,R4
55 55 10 9A 05DF MOVZBL S^#PRT$C_NONE,R5
56 D4 05E2 CLRL R6
0133 30 05E4 BSBW SETPRTSUBR
399 SETPRT RESERVE,STATUS=#SS$ IVPROTECT ;PASS RESERVED PROTECTION CODE
53 02F4 8F 3C 05E7 MOVZWL #SS$ IVPROTECT,R3
50 0000'CF DE 05EC MOVAL W^INRANGE,R0
51 0008'CF DE 05F1 MOVAL W^RETRANGE,R1
54 54 01 9A 05F6 MOVZBL S^#PRT$C_RESERVE,-4
55 55 10 9A 05F9 MOVZBL S^#PRT$C_NONE,R5
56 D4 05FC CLRL R6
0119 30 05FE BSBW SETPRTSUBR
400 NCWRITE
401 DELTVA
53 53 01 3C 060F MOVZWL S^#SS$ NORMAL,R3
50 0000'CF DE 0612 MOVAL W^INRANGE,R0
51 0008'CF DE 0617 MOVAL W^RETRANGE,R1
002B 30 061C BSBW DELTVASUBR
402 :
403 :END OF LOOP
404 :
OC 0020'CF 001C'CF F3 061F 405 AOBLEQ W^MAXPASSCNT,W^PASSCNT,160$
50 50 01 D0 0627 406 150$: MOVL #1,R0
062A 407 $EXIT_S R0
FDD4 31 0633 408 160$: BRW RSTART
0636 409

```



```

0636 411      .SBTTL  SUBROUTINES TO CALL THE SERVICES
0636 412      :
0636 413      : INPUT:
0636 414      :
0636 415      :      RO = INADR
0636 416      :      R1 = RETADR
0636 417      :      R3 = DESIRED STATUS
0636 418      :
0636 419      : OUTPUT:
0636 420      :
0636 421      :      R2 PRESERVED
0636 422      :
0636 423      CRETVASUBR:
0636 424      $CRETVA_S (R0),(R1)
51  FCE9 CF  DE 0643 425      MOVAL  W^CRETVAERR,R1      :ERROR CONTROL STRING
      14  11 0648 426      BRB      CHECK1
064A 427      :
064A 428      : INPUT:
064A 429      :
064A 430      :      RO = INADR
064A 431      :      R1 = RETADR
064A 432      :      R3 = DESIRED STATUS
064A 433      :
064A 434      : OUTPUT:
064A 435      :
064A 436      :      R2 PRESERVED
064A 437      :
064A 438      DELTVASUBR:
064A 439      $DELTVA_S (R0),(R1)
51  FCDD CF  DE 0657 440      MOVAL  W^DELTVAERR,R1      :ERROR CONTROL STRING
      00  11 065C 441      BRB      CHECK1
      53  50  D1 065E 442      CHECK1:
      4B  13 0661 443      CMPL    RO,R3      :STATUS AS DESIRED
53  0244 8F  B1 0663 444      BEQL    10$      :BRANCH IF YES
      05  12 0668 445      CMPW    #SS$_VASFULL,R3      :IF EXPECTING VIRTUAL ADDRESS SPACE
      50  1C  B1 066A 446      BNEQ    5$      :THEN EXCEEDS QUOTA MAY ALSO BE RETU
      3F  13 066D 447      CMPW    #SS$_EXQUOTA,R0
      54  DD 066F 448      BEQL    10$
54  04  AE  D0 0671 449      5$:    PUSHL   R4
      0675 450      MOVL    4(SP),R4      :ADDRESS OF ERROR
      0675 451      $FAO_S (R1),MSGLEN,MSGBUFD,R4,R0,R3,-
      06A8 452      INRANGE,INRANGE+4,RETRANGE,RETRANGE+4
      0157 30 06AA 453      POPR    #^M<R4>
      06AD 454      BSBW   TYPEMSGBUF
      06AE 455      RSB
      00D9 31 06AE 456      10$:   BRW     RANGECHK      :GO CHECK THE RETURN RANGE
      06B1 457      :
      06B1 458      : INPUT:
      06B1 459      :
      06B1 460      :
      06B1 461      :      R1 = RETADR
      06B1 462      :      R3 = DESIRED STATUS
      06B1 463      :      R4 = PAGCNT
      06B1 464      :      R5 = REGION
      06B1 465      :
      06B1 466      : OUTPUT:
      06B1 467      :

```

```

06B1 468 : R2 PRESERVED
06B1 469 :
06B1 470 EXPREGSUBR:
06B1 471 $EXPREG_S R4,(R1),R5
51 FC7C CF DE 06C0 472 MOVAL W^EXPFEGER,R1 ;ERROR CONTROL STRING
06C5 473 CHECK2:
53 50 D1 06C5 474 CMPL R0,R3 ;STATUS AS DESIRED?
39 13 06C8 475 BEQL 10$ ;BRANCH IF YES
56 04 AE DD 06CA 476 PUSHL R6
06CC 477 MOVL 4(SP),R6 ;ADDRESS OF ERROR
06D0 478 $FAO_S (R1),MSGLEN,MSGBUFD,R6,R0,R3,R4,R5,-
06D0 479 RETRANGE,RETRANGE+4
0040 8F BA 06FB 480 POPR #^M<R6>
0102 30 06FF 481 BSBW TYPEMSGBUF
05 070? 482 RSB
0000'CF 0008'CF DO 0703 483 10$: MOVL W^RETRANGE,W^INRANGE ;MAKE INPUT RANGE LOOK LIKE CRETVA/D
54 54 09 D7 070\ 484 DECL R4
0004'CF 0000'CF 54 C1 0710 485 ASHL #9,R4,R4
70 11 0718 486 ADDL3 R4,W^INRANGE,W^INRANGE+4
071A 487 BRB RANGECHK ;AND CHECK THE RETURN RANGE
071A 488 :
071A 489 : INPUT:
071A 490 :
071A 491 : R0 = INADR
071A 492 : R1 = RETADR
071A 493 : R3 = DESIRED STATUS
071A 494 : R4 = PROTECTION TO BE SET
071A 495 : R5 = DEISRED PREVIOUS PROTECTION (104 MEANS NOT SPECIFIED)
071A 496 : R6 = ADDRESS TO RETURN PREVIOUS PROTECTION
071A 497 :
071A 498 : OUTPUT:
071A 499 :
071A 500 : R2 PRESERVED
071A 501 :
071A 502 SETPRTSUBR:
071A 503 $SETPRT_S (R0),(R1),,R4,(R6)
072B 504 IFNOWRT=#1,(R6),10$ ;SKIP PREVIOUS PROTECTION CHECK
0731 505 ;IF IT WASN'T RETURNED
55 10 91 0731 506 CMPB #104,R5 ;OR IF IT WASN'T SPECIFIED
05 13 0734 507 BEQL 10$
66 55 91 0736 508 CMPB R5,(R6) ;OTHERWISE CHECK IT
05 12 0739 509 BNEQ 20$ ;AND BRANCH IF IT'S WRONG
073B 510 10$:
53 50 D1 073B 511 CMPL R0,R3 ;STATUS = DESIRED STATUS?
4A 13 073E 512 BEQL 30$ ;BRANCH IF YES
0740 513 20$:
57 04 AE DD 0740 514 PUSHL R7
0742 515 MOVL 4(SP),R7 ;ADDRESS OF ERROR
0746 516 $FAO_S SETPRTERR,MSGLEN,MSGBUFD,R7,R0,R3,-
0746 517 INRANGE,INRANGE+4,RETRANGE,RETRANGE+4,-
0746 518 PREVPROT,R5
57 BEDO 0783 519 POPL R7
007B 30 0786 520 BSBW TYPEMSGBUF
05 0789 521 RSB
078A 522 30$:
73 0010'CF 03 E1 078A 523 RANGECHK:
078A 524 BBC #CTL$V_RNGCHK,W^CTLFLG,40$ ;BRANCH IF RANGE CHECK IS DISABLED

```

```

50      70 50      E9 0790 525      BLBC  R0,40$      ;IF ERROR IN SERVICE, SKIP THE RANGE
      0000'CF 7D 0793 526      MOVQ  W^INRANCE,R0 ;RO = STARVA, R1 = ENDVA
51      51 50      D1 0798 527      CMPL  R0,R1      ;WHICH DIRECTION?
      12 50      1A 079B 528      BGTRU 10$      ;BRANCH IF BACKWARDS
      04 50      1F 079D 529      BLSSU 5$      ;BRANCH IF FORWARDS
      OC 50 1E      E0 079F 530      BBS   #30,R0,10$ ;FOR EQUAL, P0 SPACE FORWARDS, P1 BA
      07A3 531      ;
      07A3 532      ; REQUESTED RANGE IS FORWARDS
      07A3 533      ;
50      01FF 8F      AA 07A3 534 5$:  BICW  #^X1FF,R0      ;FROM BYTE 0 OF STARTVA
51      01FF 8F      AB 07A8 535      BISW  #^X1FF,R1      ;THROUGH LAST BYTE OF ENDVA
      0A 11 07AD 536      BRB   20$      ;
      07AF 537      ;
      07AF 538      ; GOING BACKWARDS IN VIRTUAL ADDRESS SPACE
      07AF 539      ;
50      01FF 8F      AB 07AF 540 10$:  BISW  #^X1FF,R0      ;LAST BYTE OF STARTVA
51      01FF 8F      AA 07B4 541      BICW  #^X1FF,R1      ;THROUGH FIRST BYTE OF ENDVA
0008'CF 50      D1 07B9 542 20$:  CMPL  R0,W^RETRANGE ;IS THIS WHAT WAS RETURNED?
      07 12 07BE 543      BNEQ  30$      ;BRANCH IF NOT, ERROR
000C'CF 51      D1 07C0 544      CMPL  R1,W^RETRANGE+4 ;THIS ONE OK TOO?
      3C 13 07C5 545      BEQL  40$      ;BRANCH IF YES, RANGE OK
      53 DD 07C7 546 30$:  PUSHL R3      ;SAVE REGISTER
53      04 AE      D0 07C9 547      MOVL  4(SP),R3      ;TO USE FOR ERROR PC
      07CD 548      $FAO_S <W^RANGERR>,MSGLEN,MSGBUFD,R3,- ;FORMAT THE ERROR MESSAGE
      07CD 549      INRANGE,INRANGE+4,RETRANGE,RETRANGE+4
      08 BA 07FE 550      POPR  #^M<R3>      ;RESTORE SAVE REGISTER
      0001 30 0800 551      BSBW  TYPMSGBUF    ;OUTPUT THE ERROR MESSAGE
      05 0803 552 40$:  RSB      ;AND RETURN

```

```

0804 554      .SBTTL MISCELLANEOUS SUBROUTINES
0804 555      :
0804 556      : TYPE A MESSAGE
0804 557      : MSGBUF IS THE ADDRESS OF THE BEGINNING OF THE STRING
0804 558      : MSGLEN CONTAINS THE SIZE (IN BYTES) OF THE STRING
0804 559      :
0804 560      TYPEMSGBUF:
0804 561      MOVL  W^MSGLEN,R0          ;SIZE TO R0
0809 562      MOVAL W^MSGBUF,R1        ;ADDRESS TO R1
080E 563      BBC   #CTL$V PIDMSG,W^CTLFLG,5$ ;BRANCH IF NO PROCESS ID REQUIRED
0814 564      MOVAL W^MSGBUFID,R1     ;ADDRESS INCLUDING PID MSG
0819 565      ADDL  S^#<MSGBUF-MSGBUFID>,R0 ;INCLUDE EXTRA BYTES IN COUNT
081C 566      5$:
081C 567      MOVL  R1,W^RAB+RAB$L_RBF   ;SET BUFFER ADDRESS
0821 568      MOVW  R0,W^RAB+RAB$W_RSZ  ;AND SIZE
0826 569      $PUT  W^RAB              ;OUTPUT THE MESSAGE
0831 570      BLBC  R0,20$
0834 571      RSB
0835 572      20$: $EXIT_S R0          ;EXIT WOTH ERROR STATUS
083E 573      :
083E 574      : INPUTS:
083E 575      :
083E 576      : 0(SP) = ADDRESS OF ERROR
083E 577      : R1 = ADDRESS OF FORMAT CONTROL STRING
083E 578      :
083E 579      : OUTPUTS:
083E 580      :
083E 581      : R2 PRESERVED
083E 582      :
083E 583      PROBERR:
083E 584      PUSHL R5
0840 585      MOVL  4(SP),R5
0844 586      $FAO_S (R1),MSGLEN,MSGBUFD,R5
085B 587      POPR_ #^M<R5>
085D 588      BSBW  TYPEMSGBUF
0860 589      RSB
0861 590
0861 591
0861 592      .END      START

```

MMGSETPRT  
Symbol table

- TEST OF \$SETPRT SYSTEM SERVICE G 13

16-SEP-1984 01:59:27 VAX/VMS Macro V04-00  
5-SEP-1984 01:58:23 [MMGTST.SRC]MMGSETPRT.MAR;1

\$\$ .TAB	= 00000074	R	02	NOWRITERR	00000368	R	03	
\$\$ .TABEND	= 00000088	R	02	NOWRITERRADR	00000238	R	03	
\$\$ .TMP	= 00000000			NOWRITERRSIZ	= 00000025			
\$\$ .TMP1	= 00000001			OUTNAMADR	00000000	R	03	
\$\$ .TMP2	= 000000CF			OUTNAMSIZ	= 0000000A			
\$\$ T1	= 00000000			PASSCNT	00000020	R	02	
\$\$ T2	= 00000004			PID	00000018	R	02	
BIT...	= 00000004			PIDCTL	00000388	R	03	52
CHECK1	0000065E	R	03	PIDCTLADR	0000032A	R	03	58
CHECK2	000006C5	R	03	PIDCTLSIZ	= 00000003			41
CRETVAERR	00000330	R	03	PIDMSG	000000D6	R	02	4C
CRETVAERRADR	0000000A	R	03	PIDMSGD	000000C4	R	02	
CRETVAERRSIZ	= 00000063			PREVPROT	00000024	R	02	21
CRETVASUBR	00000636	R	03	PROBERR	0000083E	R	03	52
CRLF	000000CC	R	02	PRTSC_NONE	= 00000010			20
CTLSM_MEMLOOP	= 00000001			PRTSC_RESERVE	= 00000001			
CTLSM_PIDMSG	= 00000004			PRTSC_TOBIG	= 00000020			
CTLSM_RNGCHK	= 00000008			PRTSC_UR	= 0000000F			
CTLSM_TSTLOOP	= 00000002			PRTSC_URSW	= 0000000C			
CTLSV_MEMLOOP	= 00000000			PRTSC_UW	= 00000004			52
CTLSV_PIDMSG	= 00000002			RAB	00000074	R	02	58
CTLSV_RNGCHK	= 00000003			RABSB_RAC	= 0000001E			41
CTLSV_TSTLOOP	= 00000001			RABSC_BID	= 00000001			4C
CTLFLG	00000010	R	02	RABSC_BLN	= 00000044			
DELTVAERR	00000338	R	03	RABSC_SEQ	= 00000000			21
DELTVAERRADR	0000006D	R	03	RABSL_CTX	= 00000018			52
DELTVAERRSIZ	= 00000063			RABSL_CBF	= 00000028			20
DELTVASUBR	0000064A	R	03	RABSL_ROP	= 00000004			
EXPREGERR	00000340	R	03	RABSW_RSZ	= 00000022			
EXPREGERRADR	000000D0	R	03	RANGECHK	0000078A	R	03	
EXPREGERRSIZ	= 00000072			RANGERR	00000370	R	03	
EXPREGSUBR	000006B1	R	03	RANGERRADR	00000260	R	03	52
FAB	00000024	R	02	RANGERRSIZ	= 0000004F			58
FABSC_BID	= 00000003			READERR	00000350	R	03	41
FABSC_BLN	= 00000050			READERRADR	000001D4	R	03	4C
FABSC_SEQ	= 00000000			READERRSIZ	= 00000021			
FABSC_VAR	= 00000002			RETRANGE	00000008	R	02	21
FABSL_ALQ	= 00000010			RSTART	0000040A	R	03	52
FABSL_FOP	= 00000004			RUN1_MSG	00000380	R	03	20
FABSV_CHAN_MODE	= 00000002			RUN1_MSGADR	000002E8	R	03	
FABSV_FILE_MODE	= 00000004			RUN1_MSGSIZ	= 00000042			
FABSV_LNM_MODE	= 00000000			SAVEAD	00000014	R	02	
FABSV_PUT	= 00000000			SETPRTERR	00000348	R	03	
FABSW_GBC	= 00000048			SETPRTERRADR	00000142	R	03	52
IDMSG	00000378	R	03	SETPRTERRSIZ	= 00000092			58
IDMSGADR	000002AF	R	03	SETPRTSUBR	0000071A	R	03	41
IDMSGSIZ	= 00000039			SIZ...	= 00000001			4C
INRANGE	00000000	R	02	SS\$_ACCVIO	= 0000000C			
MAXPASSCNT	0000001C	R	02	SS\$_EXQUOTA	= 00000017			21
MSGBUF	000000DA	R	02	SS\$_IVPROTECT	= 000002F4			52
MSGBUFD	000000BC	R	02	SS\$_LENVIO	= 0000018C			20
MSGBUFID	000000CC	R	02	SS\$_NOPRIV	= 00000024			
MSGBUFSIZ	= 000000A0			SS\$_NORMAL	= 00000001			
MSGLEN	00000088	R	02	SS\$_PAGOWNVIO	= 000001EC			
NOREADERR	00000358	R	03	SS\$_VASFULL	= 00000244			
NOREADERRADR	000001F5	R	03	START	00000390	R	03	52
NOREADERRSIZ	= 00000024			SYSSCONNECT	*****	GX	03	58

MMGSETPRT  
Symbol table

- TEST OF \$SETPRT SYSTEM SERVICE H 13

16-SEP-1984 01:59:27 VAX/VMS Macro V04-00 Page 19  
5-SEP-1984 01:58:23 [MMGTST.SRC]MMGSETPRT.MAR;1 (9)

SYSSCRETVA	*****	GX	03
SYSSDELIVA	*****	GX	03
SYSSEXIT	*****	GX	03
SYSSEXPREG	*****	GX	03
SYSSFAO	*****	X	03
SYSSOPEN	*****	GX	03
SYSSPUT	*****	GX	03
SYSSRESUME	*****	GX	03
SYSSSETPRT	*****	GX	03
TYPMSGBUF	00000804	R	03
WRITERR	00000360	R	03
WRITERRADR	00000219	R	03
WRITERRSIZ	= 00000022		

! 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
DATAO	0000017A ( 378.)	02 ( 2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC PAGE
CODE	00000861 ( 2145.)	03 ( 3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC PAGE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	19	00:00:00.11	00:00:02.18
Command processing	123	00:00:00.77	00:00:06.57
Pass 1	336	00:00:11.81	00:00:42.73
Symbol table sort	0	00:00:01.19	00:00:03.96
Pass 2	135	00:00:02.73	00:00:09.85
Symbol table output	16	00:00:00.12	00:00:00.15
Psect synopsis output	1	00:00:00.02	00:00:00.33
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	632	00:00:16.75	00:01:05.78

The working set limit was 1350 pages.  
70956 bytes (139 pages) of virtual memory were used to buffer the intermediate code.  
There were 50 pages of symbol table space allocated to hold 874 non-local and 18 local symbols.  
592 source lines were read in Pass 1, producing 20 object records in Pass 2.  
50 pages of virtual memory were used to define 42 macros.

! Macro library statistics !

Macro library name	Macros defined
-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	1
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2	27
TOTALS (all libraries)	28

MMGSETPRT  
VAX-11 Macro Run Statistics

- TEST OF \$SETPRT SYSTEM SERVICE I 13

16-SEP-1984 01:59:27 VAX/VMS Macro V04-00  
5-SEP-1984 01:58:23 [MMGTST.SRC]MMGSETPRT.MAR;1

Page 20  
(9)

1146 GETS were required to define 28 macros.

There were no errors, warnings or information messages.

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

MM  
V0

4F  
4E

45  
54  
2F

52  
4F

20  
41  
21

4F

43  
53  
4C  
44  
53  
4C  
49  
53  
4C

4E  
4C  
58

21  
45  
20

4E  
56  
37  
50

0236 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

This image shows a grid of 100 terminal windows, arranged in 10 rows and 10 columns. Each window displays a different system utility or report, with titles such as MMGEXPCNT LIS, MMGNSFWSL LIS, MMGRTDEL LIS, MMGRTFIL LIS, MMGLKWULW LIS, MMGXQUOTA LIS, MMGSETPRT LIS, and MOM. The content within each window is dense and appears to be a mix of text, tables, and possibly some graphical elements like bar charts. The overall appearance is that of a multi-processor system interface from the early 1980s.