


```
MM      MM  MM      MM      GGGGGGGG   AAAAAA   DDDDDDDD   JJ  WW  WW  SSSSSSSS  LL
MM      MM  MM      MM      GGGGGGGG   AAAAAA   DDDDDDDD   JJ  WW  WW  SSSSSSSS  LL
MMMM   MMMM  MMMM   MMMM   GG          AA      AA   DD      DD   JJ  WW  WW  SS          LL
MMMM   MMMM  MMMM   MMMM   GG          AA      AA   DD      DD   JJ  WW  WW  SS          LL
MM  MM  MM  MM  MM  MM  GG          AA      AA   DD      DD   JJ  WW  WW  SS          LL
MM  MM  MM  MM  MM  MM  GG          AA      AA   DD      DD   JJ  WW  WW  SS          LL
MM      MM  MM      MM      GG          AA      AA   DD      DD   JJ  WW  WW  SS          LL
MM      MM  MM      MM      GG          AA      AA   DD      DD   JJ  WW  WW  SSSSSS    LL
MM      MM  MM      MM      GG          AA      AA   DD      DD   JJ  WW  WW  SSSSSS    LL
MM      MM  MM      MM      GG  GGGGGG  AAAAAAAAAA DD      DD   JJ  JJ  WW  WW  SS          LL
MM      MM  MM      MM      GG  GGGGGG  AAAAAAAAAA DD      DD   JJ  JJ  WW  WW  SS          LL
MM      MM  MM      MM      GG          GG      AA      AA   DD      DD   JJ  JJ  WWW  WWW  SS          LL
MM      MM  MM      MM      GG          GG      AA      AA   DD      DD   JJ  JJ  WWW  WWW  SS          LL
MM      MM  MM      MM      GGGGGG   AA      AA   DDDDDDDD  JJJJJJ  WW  WW  SSSSSSSS  LLLLLLLLLL  ....
MM      MM  MM      MM      GGGGGG   AA      AA   DDDDDDDD  JJJJJJ  WW  WW  SSSSSSSS  LLLLLLLLLL  ....
```

```
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
LLLLLLLLLLLL IIIIII  SSSSSSSS
LLLLLLLLLLLL IIIIII  SSSSSSSS
```

(2)	50	DECLARATIONS
(2)	54	MACROS
(4)	191	DATA STORAGE AND MESSAGE STRINGS
(7)	321	INITIALIZATION
(8)	370	ADD TO WORKING SET TEST
(9)	401	ADJWSL FORCE ERRORS
(11)	412	SUBROUTINES TO CALL THE SERVICES
(18)	591	MISCELLANEOUS SUBROUTINES

```
0000 1 :  
0000 2 : MEMORY MANAGEMENT SERVICES TEST #4  
0000 3 :  
0000 4 :  
0000 5 : .TITLE MMGADJWSL - TEST OF $ADJWSL SYSTEM SERVICE  
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 Peliability 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 SHZ0002 Stephen Zalewski 20-Aug-1980  
0000 43 : Incorporated this program into MMG test package.  
0000 44 :  
0000 45 : V02-012 TSC0003 Thomas Clark 25-Jul-1980  
0000 46 : Added further tests to system services tested in this  
0000 47 : program.  
0000 48 :
```

```

0000 50      .SBTTL  DECLARATIONS
0000 51      :
0000 52      : INCLUDE FILES:
0000 53      :
0000 54      : .SBTTL  MACROS
0000 55      :
0000 56      : MACROS:
0000 57      :
0000 58      .MACRO  LIST
0000 59      .LIST   MEB
0000 60      .ENDM   LIST
0000 61
0000 62      .MACRO  NLIST
0000 63      .NLIST  MEB
0000 64      .ENDM   NLIST
0000 65
0000 66      .MACRO  ADJWSL  PAGCNT, LIMIT=#0, WSETLM, STATUS=S^#SS$ _NORMAL
0000 67      LIST
0000 68      MOVZWL  STATUS, R3
0000 69      MOVL    PAGCNT, R4
0000 70      MOVL    LIMIT, R5
0000 71      .IF     B, WSETLM
0000 72      .IF     DIF, <LIMIT>, <#0>
0000 73      MOVAL   W^WRKSETLIM, R6
0000 74      .IFF
0000 75      CLRL    R6
0000 76      .ENDC
0000 77      .IFF
0000 78      MOVAL   WSETLM, R6
0000 79      .ENDC
0000 80      BSBW    ADJWSLSUBR
0000 81      NLIST
0000 82      .ENDM   ADJWSL
0000 83
0000 84      .MACRO  CRETVA  STARTVA, ENDVA, STATUS=S^#SS$ _NORMAL, -
0000 85      LIST      INADR=W^INRANGE, RETADR=W^RETRANGE
0000 86
0000 87      .IF     NB, STARTVA
0000 88      MOVL   STARTVA, W^INRANGE
0000 89      .ENDC
0000 90      .IF     NB, ENDVA
0000 91      MOVL   ENDVA, W^INRANGE+4
0000 92      .ENDC
0000 93      MOVZWL  STATUS, R3
0000 94      MOVAL   INADR, R0
0000 95      MOVAL   RETADR, R1
0000 96      BSBW    CRETVASUBR
0000 97      NLIST
0000 98      .ENDM   CRETVA
0000 99
0000 100     .MACRO  DELTVA  STARTVA, ENDVA, ST..TUS=S^#SS$ _NORMAL, -
0000 101     LIST      INADR=W^INRANGE, RETADR=W^RETRANGE
0000 102
0000 103     .IF     NB, STARTVA
0000 104     MOVL   STARTVA, W^INRANGE
0000 105     .ENDC
0000 106     .IF     NB, ENDVA

```

```

0000 107          MOVL   ENDVA,W^INRANGE+4
0000 108          .ENDC
0000 109          MOVZWL STATUS,R3
0000 110          MOVAL  INADR,R0
0000 111          MOVAL  RETADR,R1
0000 112          BSBW   DELTVA$UBR
0000 113          NLIST
0000 114          .ENDM  DELTVA
0000 115
0000 116          .MACRO EXPREG  PAGCNT,REGION=#0,STATUS=S^#SS$_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 LKWSET  STARTVA,ENDVA,STATUS=S^#SS$_WASCLR,-
0000 132          LIST    INADR=W^INRANGE,RETADR=W^RETRANGE
0000 133
0000 134          .IF    NB,STARTVA
0000 135          MOVL   STARTVA,W^INRANGE
0000 136          .ENDC
0000 137          .IF    NB,ENDVA
0000 138          MOVL   ENDVA,W^INRANGE+4
0000 139          .ENDC
0000 140          MOVZWL STATUS,R3
0000 141          MOVAL  INADR,R0
0000 142          MOVAL  RETADR,R1
0000 143          BSBW   LKWSETSUBR
0000 144          NLIST
0000 145          .ENDM  LKWSET
0000 146
0000 147          .MACRO ULWSET  STARTVA,ENDVA,STATUS=S^#SS$_WASET,-
0000 148          LIST    INADR=W^INRANGE,RETADR=W^RETRANGE
0000 149
0000 150          .IF    NB,STARTVA
0000 151          MOVL   STARTVA,W^INRANGE
0000 152          .ENDC
0000 153          .IF    NB,ENDVA
0000 154          MOVL   STARTVA,W^INRANGE+4
0000 155          .ENDC
0000 156          MOVZWL STATUS,R3
0000 157          MOVAL  INADR,R0
0000 158          MOVAL  RETADR,R1
0000 159          BSBW   ULWSETSUBR
0000 160          NLIST
0000 161          .ENDM  ULWSET
0000 162
0000 163          .MACRO RANGECHK ONOROFF

```

```
0000 164      LIST
0000 165      .IF      IDN <ONOROFF>,<OFF>
0000 166      BICL      #CTLSM_RNGCHK,W^CTLFLG
0000 167      .IFF
0000 168      BISL      #CTLSM_RNGCHK,W^CTLFLG
0000 169      .ENDC
0000 170      NLIST
0000 171      .ENDM      RANGECHK
```

```
0000 173 :  
0000 174 : EQUATED SYMBOLS:  
0000 175 :  
0000 176 : $SSDEF  
0000 177 : $SECDEF  
0000 178 : $PRTDEF  
0000 179 : $GBLINI  
0000 180 : $VIELD CTL,0,<-  
0000 181 : <MEMLOOP,,MASK>,-  
0000 182 : <TSTLOOP,,MASK>,-  
0000 183 : <PIDMSG,,MASK>,-  
0000 184 : <RNGCHK,,MASK>-  
0000 185 :  
00000010 0000 186 : PRT$C_NONE=1@4  
0000 187 :  
0000 188 : OWN STORAGE:  
0000 189 :
```

```
: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 191 .SBTTL DATA STORAGE AND MESSAGE STRINGS
00000000 192 .PSECT DATA,PAGE,WRT,NOEXE
0000 193 INRANGE:
00000008 0000 194 .BLKL 2
0000 0008 195 RETRANGE:
00000010 0008 196 .BLKL 2
00000006 0010 197 CTLFLG: .LONG CTL$M_TSTLOOP'CTL$M_PIDMSG
00000018 0014 198 SAVEND: .BLKL 1
0000001C 0018 199 PID: .BLKL 1 ;PROCESS ID
001C 200 MAXPASSCNT: ;NUMBER OF PASSES TO RUN
00000003 001C 201 .LONG 3
0020 202 PASSCNT: ;PASS COUNTER
00000024 0020 203 .BLKL 1
0024 204 WRKSETLIM: ;RETURNED NEW WORKING SET LIMIT
00000028 0024 205 .BLKL 1
0028 206 WRKSETDEF: ;DEFAULT, MAX, MIN WORKING SET LIMIT
00000040 0028 207 .BLKL 6
00000030 0040 208 WRKSETMAX=WRKSETDEF+8
00000038 0040 209 WRKSETMIN=WRKSETDEF+16
0040 210 WRKSETMAXADD:
00000044 0040 211 .BLKL 1 ;WRKSETMAX-WRKSETDEF
0044 212 FAB: $FAB FAC=PUT, FNA=OUTNAMADR, FNS=OUTNAMSIZ ;FAB FOR OUTPUT
0094 213 RAB: $RAB FAB=FAB ;RECORD ACCESS BLOCK FOR OUTPUT
000000DC 00D8 214 MSGLEN: .BLKL 1 ;RETURN LENGTH FROM FAO
000000FA'000000A0' 00DC 215 MSGBUFD: .LONG MSGBUFSIZ,MSGBUF ;MESSAGE BUFFER DESCRIPTOR
00E4 216 PIDMSGD: .LONG MSGBUF-PIDMSG,PIDMSG
000000F6'00000004' 00E4 217 .LONG
00EC 218 ;
00EC 219 ; ***** DO NOT SEPARATE OR REORDER THE FOLLOWING LINES
00EC 220 ;
00EC 221 MSGBUFID:
00EC 222 CRLF: .BYTE ^015,^012
20 53 53 45 43 4F 52 50 00EE 223 .ASCII $PROCESS $
20 20 20 20 00F6 224 PIDMSG: .ASCII $ $
0000019A 00FA 225 MSGBUF: .BLKB 160 ;MESSAGE BUFFER USED BY FAO
000000A0 019A 226 MSGBUFSIZ=-MSGBUF
019A 227 ;
019A 228 ; ***** DO NOT SEPARATE OR REORDER THE PRECEEDING LINES
019A 229 ;

```

```

00000000 231      .PSECT CODE,PAGE,NOVRT,EXE
0000      232
0000      233 OUTNAMADR:
54 55 50 54 55 4F 24 53 59 53 0000      234 .ASCII /SYSS$OUTPUT/
0000000A 000A      235 OUTNAMSIZ=-.OUTNAMADR
000A      236
000A      237 CRETVAERRADR:
52 52 45 20 41 56 54 45 52 43 2F 21 000A      238 .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      239 .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 006D      240 CRETVAERRSIZ=-.CRETVAERRADR
006D      241
006D      242 DELTVAERRADR:
52 52 45 20 41 56 54 4C 45 44 2F 21 006D      243 .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      244 .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 00D0      245 DELTVAERRSIZ=-.DELTVAERRADR
00D0      246
00D0      247 LKWSETERRADR:
52 52 45 20 54 45 53 57 48 4C 2F 21 00D0      248 .ASCII $!/LKWSET ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
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      249 .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 0133      250 LKWSETERRSIZ=-.LKWSETERRADR
0133      251
0133      252 ULWSETERRADR:
52 52 45 20 54 45 53 57 4C 55 2F 21 0133      253 .ASCII $!/ULWSET ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
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 014B
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 016B      254 .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 0196      255 ULWSETERRSIZ=-.ULWSETERRADR
0196      256
0196      257 EXPREGERRADR:
52 52 45 20 47 45 52 50 58 45 2F 21 0196      258 .ASCII $!/EXPREG ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
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
20 3D 20 54 4E 43 47 41 50 09 2F 21 01CE
20 4E 4F 49 47 45 52 20 2C 4C 53 21 01DA
45 43 41 50 53 20 42 55 21 50 20 3D 01E6
      20 2C 01F2
4C 58 21 20 3D 20 52 44 41 54 45 52 01F4
      2F 21 4C 58 21 20 2D 20 0200
      00000072 0208
      0208 261
      0208 262
      0208 263
52 45 20 20 4C 53 57 4A 44 41 2F 21 0208
21 20 3D 20 43 50 20 2D 20 52 4F 52 0214
57 20 53 55 54 41 54 53 20 2C 4C 58 0220
55 4F 48 53 20 2C 4C 58 21 20 53 41 022C
      4C 58 21 20 45 42 20 44 4C 0238
20 3D 20 54 4E 43 47 41 50 09 2F 21 0241
57 20 54 49 4D 49 4C 20 2C 4C 53 21 024D
55 4F 48 53 20 2C 57 55 21 20 53 41 0259
      2F 21 57 55 21 20 45 42 20 44 4C 0265
      00000068 0270
      0270 266
      0270 267
45 53 20 47 4E 49 4B 52 4F 57 2F 21 0270
45 4B 43 4F 4C 2F 45 5A 49 53 20 54 027C
      20 20 3A 53 54 49 4D 49 4C 20 44 0288
2F 4C 55 21 20 54 4C 55 41 46 45 44 0293
4C 55 21 20 58 41 4D 20 2C 4C 55 21 029F
55 21 20 4E 49 4D 20 2C 4C 55 21 2F 02AB
      2F 21 4C 55 21 2F 4C 02B7
      0000004E 02BE
      02BE 271
      02BE 272
4E 41 52 20 4E 52 55 54 45 52 2F 21 02BE
4C 20 2D 20 52 4F 52 52 45 20 45 47 02CA
58 21 20 3D 20 4E 4F 49 54 41 43 4F 02D6
      4C 02E2
21 20 3D 20 52 44 41 4E 49 09 2F 21 02E3
45 52 20 2C 4C 58 21 20 2D 20 4C 58 02EF
2D 20 4C 58 21 20 3D 20 52 44 41 54 02FB
      2F 21 4C 58 21 20 0307
      0000004F 030D
      030D 276
      030D 277
4E 41 4D 20 59 52 4F 4D 45 4D 2F 21 030D
56 52 45 53 20 54 4E 45 4D 45 47 41 0319
34 23 20 54 53 45 54 20 53 45 43 49 0325
50 20 2C 29 4C 53 57 4A 44 41 28 20 0331
      2F 21 4C 55 21 20 53 53 41 033D
      00000039 0346
      0346 280
      0346 281
20 20 2A 2A 2A 2A 2A 20 20 20 2F 21 0346
4E 20 4C 4C 49 57 20 54 53 45 54 20 0352
53 55 20 4E 55 52 20 45 42 20 57 4F 035E
56 20 4C 41 4D 52 4F 4E 20 47 4E 49 036A

```

```

259 .ASCII $!/ PAGCNT = !SL, REGION = P!UB SPACE, $
260 .ASCII $RETADR = !XL - !XL!/$
261 EXPREGERRSIZ=-EXPREGERRADR
262
263 ADJWSLERRADR:
264 .ASCII $!/ADJWSL ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XLS$
265 .ASCII $!/ PAGCNT = !SL, LIMIT WAS !UW, SHOULD BE !UW!/$
266 ADJWSLERRSIZ=-ADJWSLERRADR
267
268 WSETLMCTLADR:
269 .ASCII $!/WORKING SET SIZE/LOCKED LIMITS: $
270 .ASCII $DEFAULT !UL!/UL, MAX !UL!/UL, MIN !UL!/UL!/$
271 WSETLMCTLSIZ=-WSETLMCTLADR
272
273 RANGERRADR:
274 .ASCII $!/RETURN RANGE ERROR - LOCATION = !XLS$
275 .ASCII $!/ INADR = !XL - !XL, RETADR = !XL - !XL!/$
276 RANGERRSIZ=-RANGERRADR
277
278 IDMSGADR:
279 .ASCII $!/MEMORY MANAGEMENT SERVICES TEST #4 (ADJWSL), PASS !UL!/$
280 IDMSGSIZ=-IDMSGADR
281
282 RUN1_MSGADR:
283 .ASCII $!/ ***** TEST WILL NOW BE RUN USING NORMAL VA SPACE *****$

```

MMGADJWSL
V04-000

- TEST OF \$ADJWSL SYSTEM SERVICE L 15
DATA STORAGE AND MESSAGE STRINGS

16-SEP-1984 02:01:53
5-SEP-1984 01:57:53

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

Page 9
(5)

2A 2A 20 20 20 45 43 41 50 53 20 41 0376
2A 2A 2A 0382
20 2F 21 0385
00000042 0388
0388
0388
4C 55 21 0388
00000003 0388
0388
0388

284
285
286
287
288
289
290

.ASCII \$!/ \$
RUN1_MSGSIZ=-.RUN1_MSGADR

PIDCTLADR:
.ASCII \$!UL\$
PIDCTLSIZ=-.PIDCTLADR

```
038B 292 :  
038B 293 : STRING DESCRIPTORS  
038B 294 :  
038B 295 .ALIGN LONG  
038C 296  
038C 297 CRETVAERR:  
0000000A'00000063 038C 298 .LONG CRETVAERRSIZ,CRETVAERRADR  
038C 299 DELTVAERR:  
0000006D'00000063 0394 300 .LONG DELTVAERRSIZ,DELTVAERRADR  
039C 301 EXPREGERR:  
00000196'00000072 039C 302 .LONG EXPREGERRSIZ,EXPREGERRADR  
03A4 303 LKWSETERR:  
000000D0'00000063 03A4 304 .LONG LKWSETERRSIZ,LKWSETERRADR  
03AC 305 ULWSETERR:  
00000133'0C000063 03AC 306 .LONG ULWSETERRSIZ,ULWSETERRADR  
03B4 307 ADJWSLERR:  
00000208'00000068 03B4 308 .LONG ADJWSLERRSIZ,ADJWSLERRADR  
03BC 309 WSETLMCTL:  
00000270'0000004E 03BC 310 .LONG WSETLMCTLSIZ,WSETLMCTLADR  
03C4 311 RANGERR:  
000002BE'0000004F 03C4 312 .LONG RANGERRSIZ,RANGERRADR  
03CC 313 IDMSG:  
0000030D'00000039 03CC 314 .LONG IDMSGsiz,IDMSGADR  
03D4 315 RUN1_MSG:  
00000346'00000042 03D4 316 .LONG RUN1_MSGSIZ,RUN1_MSGADR  
03DC 317 PIDCTL:  
00000388'00000003 03DC 318 .LONG PIDCTLSIZ,PIDCTLADR  
03E4 319
```

```

03E4 321          .SBTTL  INITIALIZATION
03E4 322          :*****
03E4 323          :PROGRAM DESCRIPTION:
03E4 324          :
03E4 325          : THIS PROGRAM TESTS THE FOLLOWING SYSTEM SERVICES:
03E4 326          :     $ADJWSL
03E4 327          :
03E4 328          : THE PROGRAM DOES SEVERAL WORKING SET LIMIT ADJUSTMENTS TO CHECK
03E4 329          : THAT THE SERVICE PERFORMS CORRECTLY. FOLLOWING THIS THE PROGRAM
03E4 330          : FORCES POSSIBLE ERROR PATHS FOR THE ABOVE MENTIONED SYSTEM SERVICE.
03E4 331          : THREE PASSES ARE MADE THROUGH THE TEST LOOP TO ENSURE PATH
03E4 332          : REPEATABILITY. ONLY REGULAR VA SPACE IS USED IN THIS TEST PROGRAM
03E4 333          : AS THE ADJWSL SYSTEM SERVICE IS NOT AFFECTED BY MEMORY ALLOCATION.
03E4 334          :
03E4 335          : REFER TO MASD$:[MMGTST.COM]MMGTST.RAP FOR FURTHER INFORMATION
03E4 336          : REGARDING JUST HOW COMPLETELY THE ABOVE MENTIONED SYSTEM SERVICES
03E4 337          : ARE TESTED BY THIS PROGRAM.
03E4 338          :
03E4 339          :*PRIVILEGES:
03E4 340          : THIS PROGRAM NEEDS NO SPECIAL PRIVILEGES TO EXECUTE.
03E4 341          :*****
03E4 342          :
03E4 343          : START HERE
03E4 344          :
0000 03E4 345 START:  .WORD  0          ;ENTRY MASK
03E6 346          $OPEN  W^FAB          ;OPEN THE FILE "$OUTPUT"
03F1 347          BLBC   RO,10$        ;BRANCH IF ERROR
03F4 348          $CONNECT W^RAB        ;CONNECT THE RECORD ACCESS BLOCK
03FF 349          BLBS   RO,20$
0402 350 10$:    $EXIT_S RO          ;EXIT WITH STATUS IN RO
040B 351 20$:    MOVL   #1,PASSCNT      ;INITIALIZE THE PASS COUNT
0412 352          $RESUME_S PID        ;SET UP PROCESS ID
0421 353          MOVZWL PID,RO
0428 354          $FAO_S  PIDCTL,MSGLEN,PIDMSGD,RO ;INIT THE PROCESS ID STRING
0440 355          :
0440 356          : INFORM OPERATOR THAT TESTS WILL BE RUN USING ONLY NORMAL VA SPACE
0440 357          :
0440 358          $FAO_S  RUN1 MSG,MSGLEN,MSGBUFD ;INFORM OPR NORMAL VA USED FOR TEST
0456 359          BSBW   TYPEMSGBUF
0459 360          BICL   #CTL$M_PIDMSG,W^CTLFLG ;STOP PROCESS ID FROM PRINTING
045E 361 RSTART:
045E 362 RANGECHK ON
0463 363          $FAO_S  IDMSG,MSGLEN,MSGBUFD,PASSCNT
0480 364          BSBW   TYPEMSGBUF
0483 365          EXPREG #1
0483          MOVZWL  S^#SS$_NORMAL,R3
0486          MOVL   #1,R4
0489          MOVAL  W^RETRANGE,R1
048E          CLRL  R5
0490          BSBW   EXPREGSUBR
0493 366          MOVQ  W^RETRANGE,R2
0498 367          MOVQ  R2,W^INRANGE
049D 368          MOVL  R2,W^SAVEND

```

B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
[
\
]
^
_
`
a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
r
s
t
u
v
w
x
y
z
0
1
2
3
4
5
6
7
8
9
-
_
/

```

04A2 370 .SBTTL ADD TO WORKING SET TEST
04A2 371 :
04A2 372 : TEST ADJWSL
04A2 373 :
04A2 374 CRETVA W^SAVEND,#1@30-1,#SS$ VASFULL
04A2 MOVL W^SAVEND,W^INRANGE
04A9 MOVL #1@30-1,W^INRANGE+4
04B2 MOVZWL #SS$ VASFULL,R3
04B7 MOVAL W^INRANGE,R0
04BC MOVAL W^RETRANGE,R1
04C1 BSBW CRETVASUBR
04C4 375 MOVQ W^RETRANGE,-(SP) ;SAVE RANGE FOR DEAD PAGE TABLE TEST
04C9 376 MOVAL W^WRKSETDEF,R10
04CE 377 ADJWSL #0,WSETLM=(R10)+ ;FIND DEFAULT WORKING SET LIMIT
04CE MOVZWL S^#SS$ _NORMAL,R3
04D1 MOVL #0,R4
04D4 MOVL #0,R5
04D7 MOVAL (R10)+,R6
04DA BSBW ADJWSLSUBR
04DD 378 BSBW MAXPAGLOCK ;DETERMINE MAXIMUM PAGES THAT MAY BE
04E0 379 ADJWSL #1@30,WSETLM=(R10)+ ;EXTEND LIMIT TO MAX
04E0 MOVZWL S^#SS$ _NORMAL,R3
04E3 MOVL #1@30,R4
04EA MOVL #0,R5
04ED MOVAL (R10)+,R6
04F0 BSBW ADJWSLSUBR
04F3 380 SUBL3 W^WRKSETDEF,W^WRKSETMAX,W^WRKSETMAXADD
04FD 381 BSBW MAXPAGLOCK ;DETERMINE MAX PAGES LOCKED
0500 382 ADJWSL #-<1@30>,WSETLM=(R10)+ ;CONTRACT LIMIT TO MIN
0500 MOVZWL S^#SS$ _NORMAL,R3
0503 MOVL #-<1@30>,R4
050A MOVL #0,R5
050D MOVAL (R10)+,R6
0510 BSBW ADJWSLSUBR
0513 383 BSBW MAXPAGLOCK ;NO .OF PAGES THAT CAN BE LOCKED
0516 384 :
0516 385 : WITH MINIMUM WORKING SET LIST, TOUCH A PAGE IN EACH PAGE TABLE
0516 386 : TO FORCE THE DEAD PAGE TABLE LOGIC
0516 387 :
0516 388 POPR #^M<R0,R1> ;RECOVER RANGE FROM CRETVA
0518 389 TOUCH_EACH_PT:
0518 390 INCL (R0)
051A 391 ACBL R1,#^X10000,R0,TOUCH_EACH_PT
0524 392
0524 393 SUBL3 W^WRKSETMIN,W^WRKSETDEF,R0
052C 394 ADJWSL R0,WSETLM=W^WRKSETLIM ;BACK TO DEFAULT SIZE
052C MOVZWL S^#SS$ _NORMAL,R3
052F MOVL R0,R4
0532 MOVL #0,R5
0535 MOVAL W^WRKSETLIM,R6
053A BSBW ADJWSLSUBR
053D 395 RANGECHK OFF
053D 396 DELTVA BICL #CTLSM RNGCHK,W^CTLFLG
0542 #1@30-1,W^SAVEND
0542 MOVL #1@30-1,W^INRANGE
054B MOVL W^SAVEND,W^INRANGE+4
0552 MOVZWL S^#SS$ _NORMAL,R3
0000'CF 0014'CF DO 04A2
0004'CF 3FFFFFFF 8F DO 04A9
53 0244 8F 3C 04B2
50 0000'CF DE 04B7
51 0008'CF DE 04BC
00EB 30 04C1
7E 0008'CF 7D 04C4 375
5A 0028'CF DE 04C9 376
04CE 377
53 01 3C 04CE
54 00 DO 04D1
55 00 DO 04D4
56 8A DE 04D7
0258 30 04DA
02E2 30 04DD 378
53 01 3C 04E0 379
54 40000000 8F DO 04E3
55 00 DO 04EA
56 8A DE 04ED
0242 30 04F0
0040'CF 0030'CF 0028'CF C3 04F3 380
02C2 30 04FD 381
0500 382
53 01 3C 0500
54 C0000000 8F DO 0503
55 00 DO 050A
56 8A DE 050D
0222 30 0510
02AC 30 0513 383
0516 384 :
0516 385 : WITH MINIMUM WORKING SET LIST, TOUCH A PAGE IN EACH PAGE TABLE
0516 386 : TO FORCE THE DEAD PAGE TABLE LOGIC
0516 387 :
03 BA 0516 388
60 D6 0518 389 TOUCH_EACH_PT:
FFF4 50 00010000 8F 51 F1 051A 391 ACBL R1,#^X10000,R0,TOUCH_EACH_PT
50 0028'CF 0038'CF C3 0524 393
0524 394
53 01 3C 052C
54 50 DO 052F
55 00 DO 0532
56 0024'CF DE 0535
01F8 30 053A
0010'CF 08 CA 053D 395
053D 396
0000'CF 3FFFFFFF 8F DO 0542
0004'CF 0014'CF DO 054B
53 01 3C 0552

```

50	0000'CF	DE	0555		MOVAL	W^INRANGE,R0
51	0008'CF	DE	055A		MOVAL	W^RETRANGE,R1
	0061	30	055F		BSBW	DELTVASUBR
			0562	397	RANGECHK	ON
0010'CF	08	C8	0562		BISL	#CTLSM RNGCHK,W^CTLFLG
			0567	398	\$FAOL_S	WSETLMCTL,MSGLEN,MSGBUFD,WRKSETDEF
	0201	30	0584	399	BSBW	TYPMSGBUF


```
0587 401 .SBTTL ADJWSL FORCE ERRORS
0587 402 :
0587 403 : FORCE ERRORS FROM ADJUST WORKING ' LIMIT
0587 404 :
0587 405 :
53 OC 3C 0587 ADJWSL #1,WSETLM=W^4,STATU #SS$_ACCVIO ;WSETLM NOT ACCESSIBLE
54 01 DO 058A MOVZWL #SS$_ACCVIO,R3
55 CO DO 058D MOVL #1,R4
56 0004'CF DE 0590 MOVL #0,R5
019D 30 0595 MOVAL W^4,R6
BSBW ADJWSLSUBR
```

OC 0020'CF	001C'CF	F3	0598	407	AOBLEQ	W^MAXPASSCNT,W^PASSCNT,160\$
	5C 01	D0	05A0	408 150\$:	MOVL	#1,RO
			05A3	409	\$EXIT_S	RO
	FEAF	31	05AC	410 160\$:	BRW	RSTART

```
05AF 412 .SBTTL SUBROUTINES TO CALL THE SERVICES
05AF 413 :
05AF 414 : INPUT:
05AF 415 :
05AF 416 : R0 = INADR
05AF 417 : R1 = RETADR
05AF 418 : R3 = DESIRED STATUS
05AF 419 :
05AF 420 : OUTPUT:
05AF 421 :
05AF 422 : R2 PRESERVED
05AF 423 :
05AF 424 CRETVASUBR:
05AF 425 $CRETVA_S (R0),(R1)
51 FDCC CF DE 05BC 426 MOVAL -W^CRETVAERR,R1 ;ERROR CONTROL STRING
3C 11 05C1 427 BRB CHECK1
05C3 428 :
05C3 429 : INPUT:
05C3 430 :
05C3 431 : R0 = INADR
05C3 432 : R1 = RETADR
05C3 433 : R3 = DESIRED STATUS
05C3 434 :
05C3 435 : OUTPUT:
05C3 436 :
05C3 437 : R2 PRESERVED
05C3 438 :
05C3 439 DELTVASUBR:
51 FDCO CF DE 05D0 440 $DELTVA_S (R0),(R1) ;ERROR CONTROL STRING
28 11 05D1 441 MOVAL -W^DELTVAERR,R1
05D5 442 BRB CHECK1
```

```

05D7 444 :
05D7 445 : INPUT:
05D7 446 :
05D7 447 :         R0 = INADR
05D7 448 :         R1 = RETADR
05D7 449 :         R3 = DESIRED STATUS
05D7 450 :
05D7 451 : OUTPUT:
05D7 452 :
05D7 453 :         R2 PRESERVED
05D7 454 :
05D7 455 LKWSETSUBR:
05D7 456         $LKWSET_S (R0), (R1)
51  FDBC CF  DE 05E4 457         MOVAL -W^LKWSETERR,R1           ;ERROR CONTROL STRING
      14      11 05E9 458         BRB      CHECK1
05EB 459 :
05EB 460 : INPUT:
05EB 461 :
05EB 462 :         R0 = INADR
05EB 463 :         R1 = RETADR
05EB 464 :         R3 = DESIRED STATUS
05EB 465 :
05EB 466 : OUTPUT:
05EB 467 :
05EB 468 :         R2 PRESERVED
05EB 469 :
05EB 470 ULWSETSUBR:
51  FDBO CF  DE 05EB 471         $ULWSET_S           (R0), (R1)
      00      11 05F8 472         MOVAL -W^ULWSETERR,R1           ;ERROR CONTROL STRING
      05FD 473         BRB      CHECK1
      05FF 474

```

```

53 50 D1 05FF 476 CHECK1:
    4B 13 05FF 477      CMPL  R0,R3          ;STATUS AS DESIRED
53 0244 8F B1 0602 478      BEQL  10$          ;BRANCH IF YES
    05 12 0604 479      CMPW  #SS$ _VASFULL,R3 ;IF EXPECTING VIRTUAL ADDRESS SPACE
50 1C B1 0609 480      BNEQ  5$
    3F 13 060B 481      CMPW  #SS$ _EXQUOTA,R0 ;THEN EXCEEDS QUOTA MAY ALSO BE RETU
54 04 AE D0 0610 482      BEQL  10$
    54 DD 0612 483 5$:    PUSHL R4
    04 AE D0 0616 484      MOVL  4(SP),R4      ;ADDRESS OF ERROR
    0616 485      $FAO_S (R1),MSGLEN,MSGBUFD,R4,R0,R3,-
    0616 486      INRANGE,INRANGE+4,RETRANGE,RETRANGE+4
    10 BA 0649 487      POPR  #^M<R4>
    013A 30 064B 488      BSBW  TYPEMSGBUF
    05 064E 489      RSB
    0069 31 064F 490 10$:
    064F 491      BRW  RANGECHK          ;GO CHECK THE RETURN RANGE

```

```
0652 493 :  
0652 494 : INPUT:  
0652 495 :  
0652 496 : R1 = RETADR  
0652 497 : R3 = DESIRED STATUS  
0652 498 : R4 = PAGCNT  
0652 499 : R5 = REGION  
0652 500 :  
0652 501 : OUTPUT:  
0652 502 :  
0652 503 : R2 PRESERVED  
0652 504 :  
0652 505 : EXPREGSUBR:  
0652 506 : $EXPREG_S R4,(R1),R5  
51 FD37 CF DE 0661 507 : MOVAL -W^EXPREGERR,R1 ;ERROR CONTROL STRING
```

			0666	509	CHECK2:			
	53	50	D1 0666	510		CMPL	R0,R3	;STATUS AS DESIRED?
		39	13 0669	511		BEQL	10\$;BRANCH IF YES
		56	DD 066B	512		PUSHL	R6	
	56	04	AE DD 066D	513		MOVL	4(SP),R6	;ADDRESS OF ERROR
				514		\$FAO,S	(R1),MSGLEN,MSGBUFD,R6,R0,R3,R4,R5,-	
				515		RETRANGE,RETRANGE+4		
	0040	8F	BA 069C	516		POPR	#^M<R6>	
		00E5	30 06A0	517		BSBW	TYPMSGBUF	
			05 06A3	518		RSB		
	0000'CF	0008'CF	D0 06A4	519	10\$:	MOVL	W^RETRANGE,W^INRANGE	;MAKE INPUT RANGE LOOK LIKE CRETVA/D
			D7 06AB	520		DECL	R4	
		54	78 06AD	521		ASHL	#9,R4,R4	
	0004'CF	54	C1 06B1	522		ADDL3	R4,W^INRANGE,W^INRANGE+4	
		00	11 06B9	523		BRB	RANGECHK	;AND CHECK THE RETURN RANGE
				524				
			06BB					

```

73 0010'CF 03 E1 06BB 526 RANGECHK:
    70 50 E9 06BB 527 BBC #CTL$V_RNGCHK,W^CTLFLG,40$ ;BRANCH IF RANGE CHECK IS DISABLED
50 0000'CF 7D 06C1 528 BLBC R0,40$ ;IF ERROR IN SERVICE, SKIP THE RANGE
    51 50 D1 06C4 529 MOVQ W^INRANGE,R0 ;R0 = STARVA, R1 = ENDVA
    12 1A C5CC 530 CMPL R0,R1 ;WHICH DIRECTION?
    04 1F 06CE 531 BGTRU 10$ ;BRANCH IF BACKWARDS
    OC 50 1E E0 06D0 532 BLSSU 5$ ;BRANCH IF FORWARDS
    06D4 533 BBS #30,R0,10$ ;FOR EQUAL, P0 SPACE FORWARDS, P1 BA
    06D4 534 :
    06D4 535 : REQUESTED RANGE IS FORWARDS
    06D4 536 :
50 01FF 8F AA 06D4 537 5$: BICW #^X1FF,R0 ;FROM BYTE 0 OF STARTVA
51 01FF 8F AB 06D9 538 BISW #^X1FF,R1 ;THROUGH LAST BYTE OF ENDVA
    OA 11 06DE 539 BRB 20$ ;
    06E0 540 :
    06E0 541 : GOING BACKWARDS IN VIRTUAL ADDRESS SPACE
    06E0 542 :
50 01FF 8F AB 06E0 543 10$: BISW #^X1FF,R0 ;LAST BYTE OF STARTVA
51 01FF 8F AA 06E5 544 BICW #^X1FF,R1 ;THROUGH FIRST BYTE OF ENDVA
0008'CF 50 D1 06EA 545 20$: CMPL R0,W^RETRANGE ;IS THIS WHAT WAS RETURNED?
    07 12 06EF 546 BNEQ 30$ ;BRANCH IF NOT, ERROR
000C'CF 51 D1 06F1 547 CMPL R1,W^RETRANGE+4 ;THIS ONE OK TOO?
    3C 13 06F6 548 BEQL 40$ ;BRANCH IF YES, RANGE OK
    53 DD 06F8 549 30$: PUSHL R3 ;SAVE REGISTER
    53 04 AE D0 06FA 550 MOVL 4(SP),R3 ;TO USE FOR ERROR PC
    06FE 551 $FA0_S <W^RANGERR>,MSGLEN,MSGBUFD,R3,- ;FORMAT THE ERROR MESSAGE
    06FE 552 INRANGE,INRANGE+4,RETRANGE,RETRANGE+4
    08 BA 072F 553 POPR #^M<R3> ;RESTORE SAVE REGISTER
    0054 30 0731 554 BSBW TYPEMSGBUF ;OUTPUT THE ERROR MESSAGE
    05 0734 555 40$: RSB ;AND RETURN

```



```

0735 557 :
0735 558 : INPUT:
0735 559 :
0735 560 : R3 = DESIRED STATUS
0735 561 : R4 = PAGCNT
0735 562 : R5 = DESIRED LIMIT
0735 563 : R6 = ADDRESS TO RETURN NEW WORKING SET LIMIT
0735 564 :
0735 565 : OUTPUT:
0735 566 :
0735 567 : R2 PRESERVED
0735 568 :
0735 569 ADJWSLSUBR:
0735 570 $ADJWSL_S R4,(R6)
0740 571 IFNOWRT #4,(R6),10$ :SKIP WORKING SET LIMIT CHECK
0746 572 :IF NEW LIMIT WAS NOT RETURNED
66 55 05 05 0746 573 TSTL R5 :ALSO SKIP THE CHECK
0748 574 BEQL 10$ :IF ZERO WAS SPECIFIED
66 55 05 05 074A 575 CMPW R5,(R6) :OTHERWISE CHECK DESIRED WORKING
074D 576 :SET LIMIT AGAINST THAT RETURNED
074D 577 BNEQ 20$ :BRANCH IF ERROR
53 50 05 05 074F 578 10$:
074F 579 CMPL R0,R3 :STATUS AS DESIRED?
33 13 0752 580 BEQL 30$ :BRANCH IF YES
0754 581 20$:
57 04 05 05 0754 582 PUSHL R7
0756 583 MOVL 4(SP),R7 :ADDRESS OF ERROR
075A 584 $FAO_S ADJWSLERR,MSGLEN,MSGBUFD,R7,R0,R3,-
075A 585 R4,WRKSETLIM,R5
57 04 05 05 0781 586 POPL R7
0001 30 0784 587 BSBW TYPEMSGBUF
0787 588 30$:
0787 589 RSB

```

```

0788 591          .SBTTL MISCELLANEOUS SUBROUTINES
0788 592          :
0788 593          : TYPE A MESSAGE
0788 594          : MSGBUF IS THE ADDRESS OF THE BEGINNING OF THE STRING
0788 595          : MSGLEN CONTAINS THE SIZE (IN BYTES) OF THE STRING
0788 596          :
0788 597          TYPEMSGBUF:
0788 598          MOVL    W^MSGLEN,R0          ;SIZE TO R0
078D 599          MOVAL   W^MSGBUF,R1        ;ADDRESS TO R1
0792 600          BBC     #CTL$V PIDMSG,W^CTLFLG,5$ ;BRANCH IF NO PROCESS ID REQUIRED
0798 601          MOVAL   W^MSGBUFID,R1      ;ADDRESS INCLUDING PID MSG
079D 602          ADDL    S^#<MSGBUF-MSGBUFID>,R0 ;INCLUDE EXTRA BYTES IN COUNT
07A0 603          5$:
07A0 604          MOVL    R1,W^RAB+RAB$RBF    ;SET BUFFER ADDRESS
07A5 605          MOVW   RO,W^RAB+RAB$R$RSZ  ;AND SIZE
07AA 606          $PUT   W^RAB                ;OUTPUT THE MESSAGE
07B5 607          BLBC   RO,20$
07B8 608          KSB
07B9 609          20$: $EXIT_S R0          ;EXIT WITH ERROR STATUS
07C2 610          :
07C2 611          : INPUT:
07C2 612          :
07C2 613          : INRANGE CONTAINS INFINITE RANGE OF ADDRESSES PREVIOUSLY CREATED
07C2 614          : RETRANGE CONTAINS A RANGE OF ADDRESSES
07C2 615          : R10 CONTAINS ADDRESS TO STORE THE # OF PAGES SPANNED BY RETRANGE
07C2 616          :
07C2 617          : OUTPUT:
07C2 618          :
07C2 619          : R10 UPD TO POINT AT NEXT LONG WORD
07C2 620          :
07C2 621          MAXPAGLOCK:
07C2 622          LKWSET STATUS=#SS$ LKWSETFUL ;LOCK AS MANY AS ALLOWED
07C2          MOVZWL #SS$ LKWSETFUL,R3
07C7          MOVAL   W^INRANGE,R0
07CC          MOVAL   W^RETRANGE,R1
07D1          BSBW    LKWSETSUBR
07D4 623          SUBL3  W^RETRANGE,W^RETRANGE+4,R0 ;NUMBER OF BYTES IN RANGE
07DC 624          BEQL   20$ ;BRANCH IF NONE LOCKED
07DE 625          ASHL  #-9,R0,R0 ;NUMBER OF PAGES -1
07E3 626          ADDL3 #1,R0,(R10)+ ;STORE NUMBER OF PAGES
07E7 627          RANGECHK OFF
07E7          BICL    #CTLSM_RNGCHK,W^CTLFLG
07EC 628          ULWSET INADR=W^RETRANGE ;UNLOCK THE ONES THAT WERE LOCKED
07EC          MOVZWL  S^#SS$ WASSET,R3
07EF          MOVAL   W^RETRANGE,R0
07F4          MOVAL   W^RETRANGE,R1
07F9          BSBW    ULWSETSUBR
07FC 629          RANGECHK ON
07FC          B:SL    #CTLSM_RNGCHK,W^CTLFLG
0801 630          RSB
0802 631          20$: CLRL  (R10)+ ;NO PAGES LOCKED
0804 632          RSB
0805 633
0805 634          .END START

```

\$\$TAB	= 00000094	R	02	MAXPASSCNT	0000001C	R	02	
\$\$TABEND	= 00000008	R	02	MSGBUF	000000FA	R	02	
\$\$TMP	= 00000000			MSGBLFD	000000DC	R	02	52
\$\$TMP1	= 00000001			MSGBUFID	000000EC	R	02	58
\$\$TMP2	= 000000CF			MSGBUFSIZ	= 000000A0			41
\$\$I1	= 00000000			MSGLEN	000000D8	R	02	40
\$\$I2	= 00000009			OUTNAMADR	00000000	R	03	
ADJWSLERR	00000384	R	03	OUTNAMSIZ	= 0000000A			21
ADJWSLERRADR	00000208	R	03	PASSCNT	00000020	R	02	52
ADJWSLERRSIZ	= 00000068			PID	00000018	R	02	20
ADJWSLSUBR	00000735	R	03	PIDCTL	000003DC	R	03	
BIT...	= 00000004			PIDCTLADR	00000388	R	03	20
CHECK1	000005FF	R	03	PIDCTLSIZ	= 00000003			57
CHECK2	00000666	R	03	PIDMSG	000000F6	R	02	4F
CRETVAERR	0000038C	R	03	PIDMSGD	000000E4	R	02	21
CRETVAERRADR	0000000A	R	03	PRTSC_NONE	= 00000010			
CRETVAERRSIZ	= 00000063			RAB	00000094	R	02	
CRETVASUBR	000005AF	R	03	RAB\$B_RAC	= 0000001E			
CRLF	000000EC	R	02	RAB\$C_BID	= 00000001			4F
CTLSM_MEMLOOP	= 00000001			RAB\$C_BLN	= 00000044			4E
CTLSM_PIDMSG	= 00000004			RAB\$C_SEQ	= 00000000			4E
CTLSM_RNGCHK	= 00000008			RAB\$L_CTX	= 00000018			
CTLSM_TSTLOOP	= 00000002			RAB\$L_RBF	= 00000028			
CTLSV_MEMLOOP	= 00000000			RAB\$L_ROP	= 00000004			
CTLSV_PIDMSG	= 00000002			RAB\$W_RSZ	= 00000022			
CTLSV_RNGCHK	= 00000003			RANGECHK	000006BB	R	03	45
CTLSV_TSTLOOP	= 00000001			RANGERR	000003C4	R	03	54
CTLFLG	00000010	R	02	RANGERRADR	000002BE	R	03	2F
DELTVAERR	00000394	R	03	RANGERRSIZ	= 0000004F			
DELTVAERRADR	0000006D	R	03	RETRANGE	00000008	R	02	
DELTVAERRSIZ	= 00000063			RSTART	0000045E	R	03	
DELTVASUBR	000005C3	R	03	RUN1_MSG	000003D4	R	03	52
EXPREGERR	0000039C	R	03	RUN1_MSGADR	00000346	R	03	4F
EXPREGERRADR	00000196	R	03	RUN1_MSGSIZ	= 00000042			
EXPREGERRSIZ	= 00000072			SAVEFD	00000014	R	02	
EXPREGSUBR	00000652	R	03	SIZ...	= 00000001			
FAB	00000044	R	02	SS\$ACCVID	= 0000000C			20
FAB\$C_BID	= 00000003			SS\$EXQUOTA	= 0000001C			41
FAB\$C_BLN	= 00000050			SS\$LKWSETFUL	= 00000194			21
FAB\$C_SEQ	= 00000000			SS\$NORMAL	= 00000001			
FAB\$C_VAR	= 00000002			SS\$VASFULL	= 00000244			
FAB\$L_ALQ	= 00000010			SS\$WASSET	= 00000009			
FAB\$L_FOP	= 00000004			START	000003E4	R	03	
FAB\$V_CHAN_MODE	= 00000002			SYSSADJWSL	*****	GX	03	
FAB\$V_FILE_MODE	= 00000004			SYSSCONNECT	*****	GX	03	4E
FAB\$V_LNM_MODE	= 00000000			SYSSCRETVA	*****	GX	03	4C
FAB\$V_PUT	= 00000000			SYSSDELTVA	*****	GX	03	58
FAB\$W_GBC	= 00000048			SYSSEXIT	*****	GX	03	
IDMSG	000003CC	R	03	SYS\$EXPREG	*****	GX	03	21
IDMSGADR	0000030D	R	03	SYS\$FAO	*****	X	03	45
IDMSGSIZ	= 00000039			SYS\$FAOL	*****	GX	03	2D
INRANGE	00000000	R	02	SYS\$LKWSET	*****	GX	03	
LKWSETERR	000003A4	R	03	SYS\$OPEN	*****	GX	03	
LKWSETERRADR	000000D0	R	03	SYS\$PUT	*****	GX	03	
LKWSETERRSIZ	= 00000063			SYS\$RESUME	*****	GX	03	
LKWSETSUBR	000005D7	R	03	SYS\$LWSET	*****	GX	03	4E
MAXPAGLOCK	000007C2	R	03	TOUCH_EACH_PT	00000518	R	03	56

MMGADJWSL
Symbol table

- TEST OF \$ADJWSL SYSTEM SERVICE

C 1

16-SEP-1984 02:01:53
5-SEP-1984 01:57:53

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

Page 25
(18)

MM
V0

TYPMSGBUF	00000788	R	03
ULWSETERR	000003AC	R	03
ULWSETERRADR	00000133	R	03
ULWSETERRSIZ	= 00000063		
ULWSETSUBR	000000EB	R	03
WRKSETDEF	00000028	R	02
WRKSETLIM	00000024	R	02
WRKSETMAX	= 00000030	R	02
WRKSETMAXADD	00000040	R	02
WRKSETMIN	= 00000038	R	02
WSETLMCTL	000003BC	R	03
WSETLMCTLADR	00000270	R	03
WSETLMCTLSIZ	= 0000004E		

31
50

20
4E
53
20
2A

! 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	0000019A (410.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC PAGE
CODE	00000805 (2053.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC PAGE

20
4E
53
4C
20

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	10	00:00:00.06	00:00:01.20
Command processing	72	00:00:00.72	00:00:04.00
Pass 1	331	00:00:11.28	00:00:40.53
Symbol table sort	0	00:00:01.23	00:00:03.23
Pass 2	139	00:00:02.66	00:00:08.19
Symbol table output	16	00:00:00.11	00:00:00.24
Psect synopsis output	2	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	572	00:00:16.10	00:00:57.43

20
4E
53
50

The working set limit was 1350 pages.
65169 bytes (128 pages) of virtual memory were used to buffer the intermediate code.
There were 50 pages of symbol table space allocated to hold 879 non-local and 18 local symbols.
634 source lines were read in Pass 1, producing 20 object records in Pass 2.
49 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:[SYS.LIB]STARLET.MLB;2	29
TOTALS (all libraries)	30

MMGADJWSL
VAX-11 Macro Run Statistics

- TEST OF SADIWSL SYSTEM SERVICE

D 1

16-SEP-1984 02:01:53
5-SEP-1984 01:57:53

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

Page 26
(18)

MM
VO

1152 GETS were required to define 30 macros.

There were no errors, warnings or information messages.

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

MMGEXPNT LIS	MMGNSFWL LIS	MMGRTDEL LIS	MMGRTDLSP LIS	MMGRTFIL LIS	MMGLKWLW LIS	MMGXQUOTA LIS	MMGSETPRT LIS
MOM MAP							MOM