

(2)	52	DECLARATIONS
(2)	56	MACROS
(3)	238	DATA STORAGE AND MESSAGE STRINGS
(6)	413	INITIALIZATION
(7)	463	FORCE ERRORS IN CRETVA
(8)	472	FORCE ERROR IN EXPREG
(9)	488	SUBROUTINES TO CAL' THE SERVICES
(10)	675	MISCELLANEOUS SUBROUTINES

```
0000 1 :  
0000 2 : MEMORY MANAGEMENT SERVICES TEST #8  
0000 3 :  
0000 4 :  
0000 5 : .TITLE MMGNSFWSL - TEST MMG SERVICES WITH LOW WSQUOTA/WSLIMIT  
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 SHZ0006 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 TSC0008 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^#SS$ _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  CNTREG  PAGCNT,REGION=#0,STATUS=S^#SS$_NORMAL,-
0000 117      RETADR=W^RETRANGE
0000 118      LIST
0000 119      MOVZWL  STATUS,R3
0000 120      MOVL    PAGCNT,R4
0000 121      MOVAL   RETADR,R1
0000 122      .IF     IDN,<REGION>,<#0>
0000 123      CLRL    R5
0000 124      .IFF
0000 125      MOVL    REGION,R5
0000 126      .ENDC
0000 127      BSBW    CNTREGSUBR
0000 128      NLIST
0000 129      .ENDM   CNTREG
0000 130
0000 131      .MACRO  CRETVA  STARTVA,ENDVA,STATUS=S^#SS$_NORMAL,-
0000 132      INADR=W^INRANGE,RETADR=W^RETRANGE
0000 133      LIST
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    CRETVASUBR
0000 144      NLIST
0000 145      .ENDM   CRETVA
0000 146
0000 147      .MACRO  DELTVA  STARTVA,ENDVA,STATUS=S^#SS$_NORMAL,-
0000 148      INADR=W^INRANGE,RETADR=W^RETRANGE
0000 149      LIST
0000 150      .IF     NB,STARTVA
0000 151      MOVL    STARTVA,W^INRANGE
0000 152      .ENDC
0000 153      .IF     NB,ENDVA
0000 154      MOVL    ENDVA,W^INRANGE+4
0000 155      .ENDC
0000 156      MOVZWL  STATUS,R3
0000 157      MOVAL   INADR,R0
0000 158      MOVAL   RETADR,R1
0000 159      BSBW    DELTVASUBR
0000 160      NLIST
0000 161      .ENDM   DELTVA
0000 162
0000 163      .MACRO  EXPREG  PAGCNT,REGION=#0,STATUS=S^#SS$_NORMAL,-
0000 164      RETADR=W^RETRANGE
0000 165      LIST
```

```

0000 166      MOVZWL STATUS,R3
0000 167      MOVL   PAGCNT,R4
0000 168      MOVAL  RETADR,R1
0000 169      .IF    IDN,<REGION>,<#0>
0000 170      CLRL   R5
0000 171      .IFF
0000 172      MOVL   REGION,R5
0000 173      .ENDC
0000 174      BSBW   EXPREGSUBR
0000 175      NLIST
0000 176      .ENDM  EXPREG
0000 177
0000 178      .MACRO LKWSET STARTVA,ENDVA,STATUS=S^#SS$ WASCLR,-
0000 179      LIST   INADR=W^INRANGE,RETADR=W^RETRANGE
0000 180
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   LKWSETSUBR
0000 191      NLIST
0000 192      .ENDM  LKWSET
0000 193
0000 194      .MACRO ULWSET STARTVA,ENDVA,STATUS=S^#SS$ WASSET,-
0000 195      LIST   INADR=W^INRANGE,RETADR=W^RETRANGE
0000 196
0000 197      .IF    NB,STARTVA
0000 198      MOVL   STARTVA,W^INRANGE
0000 199      .ENDC
0000 200      .IF    NB,ENDVA
0000 201      MOVL   ENDVA,W^INRANGE+4
0000 202      .ENDC
0000 203      MOVZWL STATUS,R3
0000 204      MOVAL  INADR,R0
0000 205      MOVAL  RETADR,R1
0000 206      BSBW   ULWSETSUBR
0000 207      NLIST
0000 208      .ENDM  ULWSET
0000 209
0000 210      .MACRO RANGECHK ONOROFF
0000 211      LIST
0000 212      .IF    IDN <ONOROFF>,<OFF>
0000 213      BICL   #CTLSM_RNGCHK,W^CTLFLG
0000 214      .IFF
0000 215      BISL   #CTLSM_RNGCHK,W^CTLFLG
0000 216      .ENDC
0000 217      NLIST
0000 218      .ENDM  RANGECHK
0000 219
0000 220      :
0000 221      : EQUATED SYMBOLS:
0000 222      :

```

```
0000 223 $SECDEF
0000 224 $SSDEF
0000 225 $PRTDEF
0000 226 $GBLINI
0000 227 $VIELD CTL,0,<-
0000 228 <MEMLOOP,,MASK>,- :DEFINE CONTROL BITS IN R3
0000 229 <TSTLOOP,,MASK>,- :LOOP IN MEMORY WRITE LOOP
0000 230 <PIDMSG,,MASK>,- :REDO ENTIRE TEST FROM TOP
0000 231 <RNGCHK,,MASK>- :PUT PROCESS ID IN EACH TYPEOUT
0000 232 > :ON IF CHECKING RETURN RANGE
0000010 0000 233 PRTSC_NONE=104
0000 234 :
0000 235 : OWN STORAGE:
0000 236 :
```



```

0000 238 .SBTTL DATA STORAGE AND MESSAGE STRINGS
00000000 239 .PSECT DATAO,PAGE,WRT,NOEXE
00000008 0000 240 INRANGE:
^008 241 .BLKL 2
00000010 0008 242 RETRANGE:
00000006 0010 243 .BLKL 2
00000018 0014 244 CTLFLG: .LONG CTL$M_TSTLOOP!CTL$M_PIDMSG
0018 245 SAVEND: .BLKL 1
0000001C 0018 246 HIGHPOADR:
00000020 001C 247 .BLKL 1 ;LAST BYTE ADDRESS IN PO SPACE
0020 248 PID: .BLKL 1 ;PROCESS ID
0000003 0020 249 MAXPASSCNT:
0024 250 .LONG 3 ;NUMBER OF PASSES TO RUN
00000028 0024 251 PASSCNT: .BLKL 1 ;PASS COUNTER
0000002C 0028 252 WRKSETLIM: .BLKL 1 ;RETURNED NEW WORKING SET LIMIT
002C 253 WRKSETDEF: .BLKL 6 ;DEFAULT, MAX, MIN WORKING SET LIMIT
00000044 002C 254 WRKSETMAX=WRKSETDEF+8
00000034 0044 255 WRKSETMIN=WRKSETDEF+16
0000003C 0044 256 WRKSETMAXADD: .BLKL 1 ;WRKSETMAX-WRKSETDEF
00000048 0044 257 PREVPROT:
0048 258 FAB: $FAB FAC=PUT, FNA=OUTNAMADR, FNS=OUTNAMSIZ ;FAB FOR OUTPUT
0098 259 RAB: $RAB FAB=FAB ;RECORD ACCESS BLOCK FOR OUTPUT
000000E0 00DC 260 MSGLEN: .BLKL 1 ;RETURN LENGTH FROM FAO
000000FE '000000A0' 00E0 261 MSGBUFD: .LONG MSGBUFSIZ,MSGBUF ;MESSAGE BUFFER DESCRIPTOR
00E8 262 PIDMSGD: .LONG MSGBUF-PIDMSG,PIDMSG
000000FA '00000004' 00E8 263 :
00F0 264 : ***** DO NOT SEPARATE OR REORDER THE FOLLOWING LINES
00F0 265 :
00F0 266 MSGBUFID:
00F0 267 CRLF: .BYTE ^015,^012
20 53 53 45 43 4F 52 50 00F2 268 .ASCII $PROCESS $
20 20 20 20 00FA 269 PIDMSG: .ASCII $ $
0000019E 00FE 270 MSGBUF: .BLKB 160 ;MESSAGE BUFFER USED BY FAO
000000A0 019E 271 MSGBUFSIZ=-MSGBUF
019E 272 :
019E 273 : ***** DO NOT SEPARATE OR REORDER THE PRECEEDING LINES
019E 274 :
019E 275 :
019E 276 :

```

```

00000000 282 .PSECT CODE,PAGE,NOWRT,EXE
0000 283
0000 284 OUTNAMADR:
54 55 50 54 55 4F 24 53 59 53 0000 285 .ASCII /SYSS$OUTPUT/
0000000A 000A 286 OUTNAMSIZ=-OUTNAMADR
000A 287
000A 288 CRETVAERRADR:
52 52 45 20 41 56 54 45 52 43 2F 21 000A 289 .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 290 .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 291 CRETVAERRSIZ=-CRETVAERRADR
006D 292
006D 293 DELTVAERRADR:
52 52 45 20 41 56 54 4C 4C 44 2F 21 006D 294 .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 295 .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 296 DELTVAERRSIZ=-DELTVAERRADR
00D0 297
00D0 298 LKWSETERRADR:
52 52 45 20 54 45 53 57 48 4C 2F 21 00D0 299 .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 300 .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 301 LKWSETERRSIZ=-LKWSETERRADR
0133 302
0133 303 ULWSETERRADR:
52 52 45 20 54 45 53 57 4C 55 2F 21 0133 304 .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 305 .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 306 ULWSETERRSIZ=-ULWSETERRADR
0196 307
0196 308 CNTREGERRADR:
52 52 45 20 47 45 52 54 4E 43 2F 21 0196 309 .ASCII $!/CNTREG ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
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 310
20 4E 4F 49 47 45 2 20 2C 4C 55 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 311
      2F 21 4C 58 21 20 2D 20 0200
      00000072 0208 312
      0208 313 CNTREGERRSIZ=-CNTREGERRADR
      0208 314 EXPREGERRADR:
52 52 45 20 47 45 52 50 58 45 2F 21 0208 315
58 21 20 3D 20 43 50 20 2D 20 52 4F 0214
41 57 20 53 55 54 41 54 53 20 2C 4C 0220
4C 55 4F 48 53 20 2C 4C 58 21 20 53 022C
      4C 58 21 20 45 42 20 44 0238
20 3D 20 54 4E 43 47 41 50 09 2F 21 0240 316
20 4E 4F 49 47 45 52 20 2C 4C 53 21 024C
45 43 41 50 53 20 42 55 21 50 20 3D 0258
      20 2C 0264
4C 58 21 20 3D 20 52 44 41 54 45 52 0266 317
      2F 21 4C 58 21 20 2D 20 0272
      00000072 027A 318
      027A 319 EXPREGERRSIZ=-EXPREGERRADR
      027A 320 ADJWSLERRADR:
52 45 20 20 4C 53 57 4A 44 41 2F 21 027A 321
21 20 3D 20 43 50 20 2D 20 52 4F 52 0286
57 20 53 55 54 41 54 53 20 2C 4C 58 0292
55 4F 48 53 20 2C 4C 58 21 20 53 41 029E
      4C 58 21 20 45 42 20 44 4C 02AA
20 3D 20 54 4E 43 47 41 50 09 2F 21 02B3 322
57 20 54 49 4D 49 4C 20 2C 4C 53 21 02BF
55 4F 48 53 20 2C 57 55 21 20 53 41 02CB
      2F 21 57 55 21 20 45 42 20 44 4C 02D7
      00000068 02E2 323
      02E2 324 ADJWSLERRSIZ=-ADJWSLERRADR
      02E2 325 WSETLMCTLADR:
45 53 20 47 4E 49 48 52 4F 57 2F 21 02E2 326
45 48 43 4F 4C 2F 45 5A 49 53 20 54 02EE
20 20 3A 53 54 49 4D 49 4C 20 44 02FA
2F 4C 55 21 20 54 4C 55 41 46 45 44 0305 327
4C 55 21 20 58 41 4D 20 2C 4C 55 21 0311
55 21 20 4E 49 4D 20 2C 4C 55 21 2F 031D
      2F 21 4C 55 21 2F 4C 0329
      0000004E 0330 328
      0330 329 WSETLMCTLSIZ=-WSETLMCTLADR
      0330 330 READERRADR:
4F 52 52 45 20 44 41 45 52 20 2F 21 0330 331
4E 4F 49 54 41 43 4F 4C 20 2D 20 52 033C
      2F 21 20 4C 58 21 20 3D 20 0348
      00000021 0351 332
      0351 333 READERRSIZ=-READERRADR
      0351 334 NOREADERRADR:
45 20 44 41 45 52 2D 4F 4E 20 2F 21 0351 335
54 41 43 4F 4C 20 2D 20 52 4F 52 52 035D
2F 21 20 4C 58 21 20 3D 20 4E 4F 49 0369

```

```

      .ASCII $!/ PAGCNT = !UL, REGION = P!UB SPACE, $
      .ASCII $RETADR = !XL - !XL!/$
CNTREGERRSIZ=-CNTREGERRADR
EXPREGERRADR:
      .ASCII $!/EXPREG ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
      .ASCII $!/ PAGCNT = !SL, REGION = P!UB SPACE, $
      .ASCII $RETADR = !XL - !XL!/$
EXPREGERRSIZ=-EXPREGERRADR
ADJWSLERRADR:
      .ASCII $!/ADJWSL ERROR - PC = !XL, STATUS WAS !XL, SHOULD BE !XL$
      .ASCII $!/ PAGCNT = !SL, LIMIT WAS !UW, SHOULD BE !UW!/$
ADJWSLERRSIZ=-ADJWSLERRADR
WSETLMCTLADR:
      .ASCII $!/WORKING SET SIZE/LOCKED LIMITS: $
      .ASCII $DEFAULT !UL!/!UL, MAX !UL!/!UL, MIN !UL!/!UL!/$
WSETLMCTLSIZ=-WSETLMCTLADR
READERRADR:
      .ASCII $!/ READ ERROR - LOCATION = !XL !/$
READERRSIZ=-READERRADR
NOREADERRADR:
      .ASCII $!/ NO-READ ERROR - LOCATION = !XL !/$

```

MMG
Syn
SS.
SS.
SS.
SS.
SS.
SS.
AD.
AD.
AD.
AD.
BIT
CHE
CHE
CNT
CNT
CNT
CNT
CRE
CRE
CRE
CRE
CRI
CTL
CTL
CTL
CTL
CTL
DEL
DEL
DEL
DEL
EXP
EXP
EXP
FAE
FAE
FAE
FAE
FAE
FAE
HIC
IDI
IDI
INI

MMGNSFWSL
V04-000

- TEST MMG SERVICES WITH LOW WSQUOTA/WSL 16-SEP-1984 02:07:01 VAX/VMS Macro V04-00 Page 10
DATA STORAGE AND MESSAGE STRINGS 5-SEP-1984 01:58:18 [MMGTST.SRC]MMGNSFWSL.MAR;1 (4)

	0505	366	
	0505	367	PIDCTLADR:
4C 55 21	0505	368	.ASCII \$!UL\$
00000003	0508	369	PIDCTLSIZ=-PIDCTLADR
	0508	370	

MMG
VA)

Syn
Pse
Crc
Ass

The
655
The
738
54

Mac

-S
-S
TOI

115
The
MAC

```
0508 372 :  
0508 373 : STRING DESCRIPTORS  
0508 374 :  
0508 375 .ALIGN LONG  
0508 376  
0508 377 CRETVAERR:  
0000000A'00000063 0508 378 .LONG CRETVAERRSIZ,CRETVAERRADR  
0510 379 DELTVAERR:  
0000006D'00000063 0510 380 .LONG DELTVAERRSIZ,DELTVAERRADR  
0518 381 CNTREGERR:  
00000196'00000072 0518 382 .LONG CNTREGERRSIZ,CNTREGERRADR  
0520 383 EXPREGERR:  
00000208'00000072 0520 384 .LONG EXPREGERRSIZ,EXPREGERRADR  
0528 385 LKWSETERR:  
000000D0'00000063 0528 386 .LONG LKWSETERRSIZ,LKWSETERRADR  
0530 387 ULWSETERR:  
00000133'00000063 0530 388 .LONG ULWSETERRSIZ,ULWSETERRADR  
0538 389 ADJWSLERR:  
0000027A'00000068 0538 390 .LONG ADJWSLERRSIZ,ADJWSLERRADR  
0540 391 WSETLMCTL:  
000002E2'0000004E 0540 392 .LONG WSETLMCTLSIZ,WSETLMCTLADR  
0548 393 READERR:  
00000330'00000021 0548 394 .LONG READERRSIZ,READERRADR  
0550 395 NOREADERR:  
00000351'00000024 0550 396 .LONG NOREADERRSIZ,NOREADERRADR  
0558 397 WRITERR:  
00000375'00000022 0558 398 .LONG WRITERRSIZ,WRITERRADR  
0560 399 NOWRITERR:  
00000397'00000025 0560 400 .LONG NOWRITERRSIZ,NOWRITERRADR  
0568 401 MEMLOOPCTL:  
000003BC'0000007F 0568 402 .LONG MEMLOOPCTLSIZ,MEMLOOPCTLADR  
0570 403 RANGERR:  
0000043B'0000004F 0570 404 .LONG RANGERRSIZ,RANGERRADR  
0578 405 IDMSG:  
0000048A'00000039 0578 406 .LONG IDMSGsiz,IDMSGADR  
0580 407 RUN1_MSG:  
000004C3'00000042 0580 408 .LONG RUN1_MSGSIZ,RUN1_MSGADR  
0588 409 PIDCTL:  
00000505'00000003 0588 410 .LONG PIDCTLSIZ,PIDCTLADR  
0590 411
```

```

0590 413 .SBTTL INITIALIZATION
0590 414 :*****
0590 415 :PROGRAM DESCRIPTION:
0590 416 :
0590 417 :   THIS PROGRAM TESTS THE FOLLOWING SYSTEM SERVICES:
0590 418 :   $CRETVA, $EXPREG
0590 419 :
0590 420 :   THE PROGRAM DOES EACH OF THESE SYSTEM SERVICES WHEN
0590 421 :   THE WSQUOTA/WSLIMIT IS SET TO A MINIMUM IN ORDER TO FORCE KNOWN
0590 422 :   ERROR PATHS. THREE PASSES ARE MADE THROUGH THE TEST LOOP IN
0590 423 :   ORDER TO ENSURE PATH REPEATABILITY. ONLY REGULAR VA SPACE IS
0590 424 :   USED IN THIS PROGRAM.
0590 425 :
0590 426 :   REFER TO MASDS:[MMGTST.COM]MMGTST.RAP FOR FURTHER INFORMATION
0590 427 :   REGARDING JUST HOW COMPLETELY THE ABOVE MENTIONED SYSTEM SERVICES
0590 428 :   ARE TESTED BY THIS PROGRAM.
0590 429 :
0590 430 :*PRIVILEGES:
0590 431 :   THIS PROGRAM NEEDS NO SPECIAL PRIVILEGES TO EXECUTE.
0590 432 :*****
0590 433 :
0590 434 : START HERE
0590 435 :
0000 0590 436 START: .WORD 0 ;ENTRY MASK
OE 50 E9 0592 437 $OPEN W^FAB ;OPEN THE FILE '$OUTPUT'
09 50 E8 059D 438 BLBC R0,10$ ;BRANCH IF ERROR
00000024'EF 01 D0 05A0 439 $CONNECT W^RAB ;CONNECT THE RECORD ACCESS BLOCK
50 0000001C'EF 3C 05AB 440 BLBS R0,20$
05AF 441 10$: $EXIT_S R0 ;EXIT WITH STATUS IN R0
05B7 442 20$: MOVL #1,PASSCNT ;INITIALIZE THE PASS COUNT
05BE 443 $RESUME_S PID ;SET UP PROCESS ID
05CD 444 MOVZWL -PID,R0
05D4 445 $FAO_S PIDCTL,MSGLEN,PIDMSGD,R0 ;INIT THE PROCESS ID STRING
05EC 446 :
05EC 447 : INFORM OPERATOR THAT TESTS WILL BE RUN USING ONLY NORMAL VA SPACE
05EC 448 :
05EC 449 $FAO_S RUN1 MSG,MSGLEN,MSGBUFD ;INFORM OPR NORMAL VA USED FOR TEST
0010'CF 02B3 30 0602 450 BSBW TYPMSGBUF
0605 451 BICL #CTL$M_PIDMSG,W^CTLFLG ;STOP PROCESS ID FROM PRINTING
060A 452 RSTART:
060A 453 RANGECHK ON
060F 454 $FAO_S IDMSG,MSGLEN,MSGBUFD,PASSCNT
0289 30 062C 455 BSBW TYPMSGBUF
062F 456 EXPREG #1
53 01 3C 062F MOVZWL S^SS$_NORMAL,R3
54 01 D0 0632 MOVL #1,R4
51 0008'CF DE 0635 MOVAL W^RETRANGE,R1
55 D4 063A CLRL R5
0143 30 063C BSBW EXPREGSUBR
52 0008'CF 7D 063F 457 MOVQ W^RETRANGE,R2
0000'CF 52 7D 0644 458 MOVQ R2,W^INRANGE
0014'CF 52 D0 0649 459 MOVL R2,W^SAVEND
064E 460
064E 461

```

```

0000'CF 0014'CF DO 064E 463 .SBTTL FORCE ERRORS IN CRETVA
0004'CF 3FFFFFFF 8F DO 064E 464 :
53 011C 8F 3C 064E 465 : FORCE ERRORS FROM CRETVA
50 0000'CF DE 064E 466 :
51 0008'CF DE 064E 467 :
0000'CF 0008'CF 7D 064E 468 CRETVA W^SAVEND,#1230-1,#SS$ INSFWSL ;EXCEED WORKING SET LIMIT
53 01 3C 064E 469 MOVL W^SAVEND,W^INRANGE
50 0000'CF DE 0655 MOVL #1230-1,W^INRANGE+4
51 0008'CF DE 065E MOVZWL #SS$ INSFWSL,R3
0058 30 0663 MOVAL W^INRANGE,R0
0000'CF 0008'CF 7D 0668 MOVAL W^RETRANGE,R1
53 01 3C 066D BSBW CRETVASUBR
50 0000'CF DE 0670 468 MOVQ W^RETRANGE,W^INRANGE
51 0008'CF DE 0677 469 DELTVA ;DELETE WHAT WE CREATED
0058 30 0677 MOVZWL S^#SS$ NORMAL,R3
0000'CF DE 067A MOVAL W^INRANGE,R0
0008'CF DE 067F MOVAL W^RETRANGE,R1
0058 30 0684 BSBW DELTVASUBR
0687 470

```



```

0687 472 : .SBTTL FORCE ERROR IN EXPREG
0687 473 :
0687 474 : : FORCE ERRORS FROM EXPREG
0687 475 : :
0687 476 : EXPREG #1@21-1,#0,STATUS=#SS$ INSFWSL :EXCEED WORKING SET LIMIT
54 53 011C 8F 3C 0687 MOVZWL #SS$ INSFWSL,R3
001FFFFFF 8F D0 068C MOVL #1@2T-1,R4
51 0008'CF DE 0693 MOVAL W^RETRANGE,R1
00E5 D4 0698 CLRL R5
0000'CF 0008'CF 7D 069A BSBW EXPREGSUBR
069D 477 MOVQ W^RETRANGE,W^INRANGE
06A4 478 DELTVA :DELETE WHAT WE CREATED
53 01 3C 06A4 MOVZWL S^#SS$ NORMAL,R3
50 0000'CF DE 06A7 MOVAL W^INRANGE,R0
51 0008'CF DE 06AC MOVAL W^RETRANGE,R1
002B 30 06B1 BSBW DELTVASUBR
06B4 479 :
06B4 480 : END OF LOOP
06B4 481 :
OC 0024'CF 0020'CF F3 06B4 482 AOBLEQ W^MAXPASSCNT,W^PASSCNT,160$
50 01 D0 06BC 483 150$: MOVL #1,R0
06BF 484 $EXIT_S RO
FF3F 31 06C8 485 160$: BRW RSTART
06CB 486

```

```
06CB 488 .SBTTL SUBROUTINES TO CALL THE SERVICES
06CB 489 :
06CB 490 : INPUT:
06CB 491 :
06CB 492 : R0 = INADR
06CB 493 : R1 = RETADR
06CB 494 : R3 = DESIRED STATUS
06CB 495 :
06CB 496 : OUTPUT:
06CB 497 :
06CB 498 : R2 PRESERVED
06CB 499 :
06CB 500 CRETVASUBR:
06CB 501 $CRETVA_S (R0), (R1)
51 FE2C CF DE 06DB 502 MOVAL -W^CRETVAERR,R1 ;ERROR CONTROL STRING
3A 11 06DD 503 BRB CHECK1
06DF 504 :
06DF 505 : INPUT:
06DF 506 :
06DF 507 : R0 = INADR
06DF 508 : R1 = RETADR
06DF 509 : R3 = DESIRED STATUS
06DF 510 :
06DF 511 : OUTPUT:
06DF 512 :
06DF 513 : R2 PRESERVED
06DF 514 :
06DF 515 DELTVASUBR:
06DF 516 $DELTVA_S (R0), (R1)
51 FE20 CF DE 06EC 517 MOVAL -W^DELTVAERR,R1 ;ERROR CONTROL STRING
26 11 06F1 518 BRB CHECK1
06F3 519 :
06F3 520 : INPUT:
06F3 521 :
06F3 522 : R0 = INADR
06F3 523 : R1 = RETADR
06F3 524 : R3 = DESIRED STATUS
06F3 525 :
06F3 526 : OUTPUT:
06F3 527 :
06F3 528 : R2 PRESERVED
06F3 529 :
06F3 530 LKWSETSUBR:
06F3 531 $LKWSET_S (R0), (R1)
51 FE24 CF DE 0700 532 MOVAL -W^LKWSETERR,R1 ;ERROR CONTROL STRING
12 11 0705 533 BRB CHECK1
0707 534 :
0707 535 : INPUT:
0707 536 :
0707 537 : R0 = INADR
0707 538 : R1 = RETADR
0707 539 : R3 = DESIRED STATUS
0707 540 :
0707 541 : OUTPUT:
0707 542 :
0707 543 : R2 PRESERVED
0707 544 :
```

```

0707 545 ULWSETSUBR:
0707 546 $ULWSET_S (R0), (R1)
51 FE18 CF DE 0714 547 MOVAL -W^ULWSETERR,R1 ;ERROR CONTROL STRING
0719 548 CHECK1:
53 50 D1 0719 549 Cmpl R0,R3 ;STATUS AS DESIRED
48 13 071C 550 BEQL 10$ ;BRANCH IF YES
53 0244 8F B1 071E 551 CMPW #SS$_VAFULL,R3 ;IF EXPECTING VIRTUAL ADDRESS SPACE
05 12 0723 552 BNEQ 5$
50 1C B1 0725 553 CMPW #SS$_EXQUOTA,R0 ;THEN EXCEEDS QUOTA MAY ALSO BE RETU
3F 13 0728 554 BEQL 10$
54 04 AE DD 072A 555 5$: PUSHL R4
072C 556 MOVL 4(SP),R4 ;ADDRESS OF ERROR
0730 557 $FAO_S (R1),MSGLEN,MSGBUFD,R4,R0,R3,-
0730 558 INRANGE,INRANGE+4,RETRANGE,RETRANGE+4
i0 BA 0763 559 POPR #^M<R4>
0150 30 0765 560 BSBW TYPEMSGBUF
05 0768 561 RSB
0769 562 10$:
007F 31 0769 563 BRW RANGECHK ;GO CHECK THE RETURN RANGE
076C 564 :
076C 565 : INPUT:
076C 566 :
076C 567 : R1 = RETADR
076C 568 : R3 = DESIRED STATUS
076C 569 : R4 = PAGCNT
076C 570 : R5 = REGION
076C 571 :
076C 572 : OUTPUT:
076C 573 :
076C 574 : R2 PRESERVED
076C 575 :
076C 576 CNTREGSUBR:
51 FD99 CF DE 0778 577 $CNTREG_S R4,(R1),,R5
14 11 077B 578 MOVAL -W^CNTREGERR,R1 ;ERROR CONTROL STRING
0780 579 BRB CHECK2
0782 580 :
0782 581 : INPUT:
0782 582 :
0782 583 : R1 = RETADR
0782 584 : R3 = DESIRED STATUS
0782 585 : R4 = PAGCNT
0782 586 : R5 = REGION
0782 587 :
0782 588 : OUTPUT:
0782 589 :
0782 590 : R2 PRESERVED
0782 591 :
0782 592 EXPREGSUBR:
51 FD8B CF DE 0791 593 $EXPREG_S R4,(R1),,R5
53 50 D1 0796 594 MOVAL -W^EXPREGERR,R1 ;ERROR CONTROL STRING
39 13 0799 595 CHECK2:
56 50 DD 079B 596 Cmpl R0,R3 ;STATUS AS DESIRED?
04 AE DD 079D 597 BEQL 10$ ;BRANCH IF YES
56 04 AE DD 079B 598 PUSHL R6
079D 599 MOVL 4(SP),R6 ;ADDRESS OF ERROR
07A1 600 $FAO_S (R1),MSGLEN,MSGBUFD,R6,R0,R3,R4,R5,-
07A1 601 RETRANGE,RETRANGE+4

```

```

0040 8F BA 07CC 602 POPR #^M<R6>
00E5 30 07D0 603 BSBW TYPMSGBUF
05 07D3 604 RSB
0000'CF 0008'CF D0 07D4 605 10$: MOVL W^RETRANGE,W^INRANGE ;MAKE INPUT RANGE LOOK LIKE CRETVA/D
54 54 09 D7 07DB 606 DECL R4
0004'CF 0000'CF 54 09 78 07DD 607 ASHL #9,R4,R4
54 00 11 07E1 608 ADDL3 R4,W^INRANGE,W^INRANGE+4
00 11 07E9 609 BRB RANGECHK ;AND CHECK THE RETURN RANGE
07EB 610
07EB 611 RANGECHK:
73 0010'CF 03 E1 07EB 612 BBC #CTLSV_RNGCHK,W^CTLFLG,40$ ;BRANCH IF RANGE CHECK IS DISABLED
70 50 E9 07F1 613 BLBC R0,40$- ;IF ERROR IN SERVICE, SKIP THE RANGE
50 0000'CF 7D 07F4 614 MOVQ W^INRANGE,R0 ;R0 = STARVA, R1 = ENDVA
51 50 D1 07F9 615 CML R0,R1 ;WHICH DIRECTION?
12 1A 07FC 616 BGTRU 10$ ;BRANCH IF BACKWARDS
04 1F 07FE 617 BLSSU 5$ ;BRANCH IF FORWARDS
OC 50 1E E0 0800 618 BBS #30,R0,10$ ;FOR EQUAL, P0 SPACE FORWARDS, P1 BA
0804 619
0804 620 : REQUESTED RANGE IS FORWARDS
0804 621
50 01FF 8F AA 0804 622 5$: BICW #^X1FF,R0 ;FROM BYTE 0 OF STARTVA
51 01FF 8F AB 0809 623 BISW #^X1FF,R1 ;THROUGH LAST BYTE OF ENDVA
0A 11 080E 624 BRB 20$
0810 625
0810 626 : GOING BACKWARDS IN VIRTUAL ADDRESS SPACE
0810 627
50 01FF 8F AB 0810 628 10$: BISW #^X1FF,R0 ;LAST BYTE OF STARTVA
51 01FF 8F AA 0815 629 BICW #^X1FF,R1 ;THROUGH FIRST BYTE OF ENDVA
0008'CF 50 D1 081A 630 20$: CML R0,W^RETRANGE ;IS THIS WHAT WAS RETURNED?
07 12 081F 631 BNEQ 30$ ;BRANCH IF NOT, ERROR
000C'CF 51 D1 0821 632 CML R1,W^RETRANGE+4 ;THIS ONE OK TO?
3C 13 0826 633 BEQL 40$ ;BRANCH IF YES, RANGE OK
53 04 AE DD 0828 634 30$: PUSHL R3 ;SAVE REGISTER
08 08 082A 635 MOVL 4(SP),R3 ;TO USE FOR ERROR PC
0054 08 082E 636 $FAO_S <W^RANGERR>,MSGLEN,MSGBUFD,R3,- ;FORMAT THE ERROR MESSAGE
05 08 082E 637 INRANGE,INRANGE+4,RETRANGE,RETRANGE+4
08 08 085F 638 POPR #^M<R3> ;RESTORE SAVE REGISTER
30 0861 639 BSBW TYPMSGBUF ;OUTPUT THE ERROR MESSAGE
05 0864 640 40$: RSB ;AND RETURN
0865 641
0865 642 : INPUT:
0865 643
0865 644 : R3 = DESIRED STATUS
0865 645 : R4 = PAGCNT
0865 646 : R5 = DESIRED LIMIT
0865 647 : R6 = ADDRESS TO RETURN NEW WORKING SET LIMIT
0865 648
0865 649 : OUTPUT:
0865 650
0865 651 : R2 PRESERVED
0865 652
0865 653 ADJWSLSUBR:
0865 654 $ADJWSL S R4,(R6)
0870 655 IFNOWRT #4,(R6),10$ ;SKIP WORKING SET LIMIT CHECK
0876 656 ;IF NEW LIMIT WAS NOT RETURNED
55 D5 0876 657 TSTL R5 ;ALSO SKIP THE CHECK
05 13 0878 658 BEQL 10$ ;IF ZERO WAS SPECIFIED

```



```

0888 675      .SBTTL MISCELLANEOUS SUBROUTINES
0888 676      :
0888 677      : TYPE A MESSAGE
0888 678      : MSGBUF IS THE ADDRESS OF THE BEGINNING OF THE STRING
0888 679      : MSGLEN CONTAINS THE SIZE (IN BYTES) OF THE STRING
0888 680      :
0888 681      TYPEMSGBUF:
0888 682      MOVL  W^MSGLEN,R0      ;SIZE TO R0
0888 683      MOVAL W^MSGBUF,R1      ;ADDRESS TO R1
08 0010'CF 02  E1 08C2 684      BBC  #CTLSV PIDMSG,W^CTFLG,5$ ;BRANCH IF NO PROCESS ID REQUIRED
0888 685      MOVAL W^MSGBUFID,R1    ;ADDRESS INCLUDING PID MSG
0888 686      ADDL  S^#<MSGBUF-MSGBUFID>,R0 ;INCLUDE EXTRA BYTES IN COUNT
0888 687      5$:
0888 688      MOVL  R1,W^RAB+RAB$L_RBF ;SET BUFFER ADDRESS
0888 689      MOVW  R0,W^RAB+RAB$W_RSZ ;AND SIZE
0888 690      $PUT  W^RAB ;OUTPUT THE MESSAGE
0888 691      BLBC  R0,20$
0888 692      RSB
0888 693      20$: $EXIT_S R0 ;EXIT WITH ERROR STATUS
0888 694      :
0888 695      : INPUTS:
0888 696      :
0888 697      : 0(SP) = ADDRESS OF ERROR
0888 698      : R1 = ADDRESS OF FORMAT CONTROL STRING
0888 699      :
0888 700      : OUTPUTS:
0888 701      :
0888 702      : R2 PRESERVED
0888 703      :
0888 704      PROBERR:
0888 705      PUSHL R5
0888 706      MOVL  4(SP),R5
0888 707      $FAO_S (R1),MSGLEN,MSGBUFD,R5
0888 708      POPR  #^M<R5>
0888 709      BSBW  TYPEMSGBUF
0888 710      RSB
0888 711      :
0888 712      : INPUT:
0888 713      :
0888 714      : INRANGE CONTAINS INFINITE RANGE OF ADDRESSES PREVIOUSLY CREATED
0888 715      : RETRANGE CONTAINS A RANGE OF ADDRESSES
0888 716      : R10 CONTAINS ADDRESS TO STORE THE # OF PAGES SPANNED BY RETRANGE
0888 717      :
0888 718      : OUTPUT:
0888 719      :
0888 720      : R10 UPDATED TO POINT AT NEXT LONG WORD
0888 721      :
0888 722      :
0888 723      :
0888 724      MAXPAGLOCK:
0888 725      LKWSET STATUS=#SS$ LKWSETFUL ;LOCK AS MANY AS ALLOWED
0888 726      MOVZWL #SS$ LKWSETFUL,R3
0888 727      MOVAL  W^INRANGE,R0
0888 728      MOVAL  W^RETRANGE,R1
0888 729      BSBW  LKWSETSUBR
0888 730      SUBL3 W^RETRANGE,W^RETRANGE+4,R0 ;NUMBER OF BYTES IN RANGE
0888 731      BEQL  20$ ;BRANCH IF NONE LOCKED
0888 732      :
0888 733      :
0888 734      :
0888 735      :
0888 736      :
0888 737      :
0888 738      :
0888 739      :
0888 740      :
0888 741      :
0888 742      :
0888 743      :
0888 744      :
0888 745      :
0888 746      :
0888 747      :
0888 748      :
0888 749      :
0888 750      :
0888 751      :
0888 752      :
0888 753      :
0888 754      :
0888 755      :
0888 756      :
0888 757      :
0888 758      :
0888 759      :
0888 760      :
0888 761      :
0888 762      :
0888 763      :
0888 764      :
0888 765      :
0888 766      :
0888 767      :
0888 768      :
0888 769      :
0888 770      :
0888 771      :
0888 772      :
0888 773      :
0888 774      :
0888 775      :
0888 776      :
0888 777      :
0888 778      :
0888 779      :
0888 780      :
0888 781      :
0888 782      :
0888 783      :
0888 784      :
0888 785      :
0888 786      :
0888 787      :
0888 788      :
0888 789      :
0888 790      :
0888 791      :
0888 792      :
0888 793      :
0888 794      :
0888 795      :
0888 796      :
0888 797      :
0888 798      :
0888 799      :
0888 800      :
0888 801      :
0888 802      :
0888 803      :
0888 804      :
0888 805      :
0888 806      :
0888 807      :
0888 808      :
0888 809      :
0888 810      :
0888 811      :
0888 812      :
0888 813      :
0888 814      :
0888 815      :
0888 816      :
0888 817      :
0888 818      :
0888 819      :
0888 820      :
0888 821      :
0888 822      :
0888 823      :
0888 824      :
0888 825      :
0888 826      :
0888 827      :
0888 828      :
0888 829      :
0888 830      :
0888 831      :
0888 832      :
0888 833      :
0888 834      :
0888 835      :
0888 836      :
0888 837      :
0888 838      :
0888 839      :
0888 840      :
0888 841      :
0888 842      :
0888 843      :
0888 844      :
0888 845      :
0888 846      :
0888 847      :
0888 848      :
0888 849      :
0888 850      :
0888 851      :
0888 852      :
0888 853      :
0888 854      :
0888 855      :
0888 856      :
0888 857      :
0888 858      :
0888 859      :
0888 860      :
0888 861      :
0888 862      :
0888 863      :
0888 864      :
0888 865      :
0888 866      :
0888 867      :
0888 868      :
0888 869      :
0888 870      :
0888 871      :
0888 872      :
0888 873      :
0888 874      :
0888 875      :
0888 876      :
0888 877      :
0888 878      :
0888 879      :
0888 880      :
0888 881      :
0888 882      :
0888 883      :
0888 884      :
0888 885      :
0888 886      :
0888 887      :
0888 888      :
0888 889      :
0888 890      :
0888 891      :
0888 892      :
0888 893      :
0888 894      :
0888 895      :
0888 896      :
0888 897      :
0888 898      :
0888 899      :
0888 900      :
0888 901      :
0888 902      :
0888 903      :
0888 904      :
0888 905      :
0888 906      :
0888 907      :
0888 908      :
0888 909      :
0888 910      :
0888 911      :
0888 912      :
0888 913      :
0888 914      :
0888 915      :
0888 916      :
0888 917      :
0888 918      :
0888 919      :
0888 920      :
0888 921      :
0888 922      :
0888 923      :
0888 924      :
0888 925      :
0888 926      :
0888 927      :
0888 928      :
0888 929      :
0888 930      :
0888 931      :
0888 932      :
0888 933      :
0888 934      :
0888 935      :
0888 936      :
0888 937      :
0888 938      :
0888 939      :
0888 940      :
0888 941      :
0888 942      :
0888 943      :
0888 944      :
0888 945      :
0888 946      :
0888 947      :
0888 948      :
0888 949      :
0888 950      :
0888 951      :
0888 952      :
0888 953      :
0888 954      :
0888 955      :
0888 956      :
0888 957      :
0888 958      :
0888 959      :
0888 960      :
0888 961      :
0888 962      :
0888 963      :
0888 964      :
0888 965      :
0888 966      :
0888 967      :
0888 968      :
0888 969      :
0888 970      :
0888 971      :
0888 972      :
0888 973      :
0888 974      :
0888 975      :
0888 976      :
0888 977      :
0888 978      :
0888 979      :
0888 980      :
0888 981      :
0888 982      :
0888 983      :
0888 984      :
0888 985      :
0888 986      :
0888 987      :
0888 988      :
0888 989      :
0888 990      :
0888 991      :
0888 992      :
0888 993      :
0888 994      :
0888 995      :
0888 996      :
0888 997      :
0888 998      :
0888 999      :
0888 1000     :

```

MMG
V04
21
4F
4E
45
54
2F
52
4F
20
41
21
4E
4C
58
21
45
2D
4E
56
32
50
20
4E
53
56
2A

```

50 50 F7 8F 78 0931 728
8A 50 01 C1 0936 729
0010'CF 08 CA 093A 730
53 09 3C 093F 731
50 0008'CF DE 0942
51 0008'CF DE 0947
FDB8 30 094C
0010'CF 08 C8 094F 732
8A D4 0955 734 20$:
05 0957 735
0958 736
0958 737
0958 738

```

```

ASHL #9,RO,RO
ADDL3 #1,RO,(R10)+
RANGECHK OFF
ULWSET BICL #CTLSM_RNGCHK,W^CTLFLG
INADR=W^RETRANGE
MOVZWL S^#SS$ WASSET,R3
MOVAL W^RETRANGE,RO
MOVAL W^RETRANGE,R1
BSBW ULWSETSUBR
RANGECHK ON
BISL #CTLSM_RNGCHK,W^CTLFLG
RSB
CLRL (R10)+
RSB
.END START

```

```

:NUMBER OF PAGES -1
:STORE NUMBER OF PAGES
:UNLOCK THE ONES THAT WERE LOCKED
:NO PAGES LOCKED

```

\$\$.TAB	= 00000098	R	02	LKWSETERR	00000528	R	03
\$\$.TABEND	= 000000DC	R	02	LKWSETERRADR	000000D0	R	03
\$\$.TMP	= 00000000			LKWSETERRSIZ	= 00000063		
\$\$.TMP1	= 00000001			LKWSETSUBR	000006F3	R	03
\$\$.TMP2	= 000000CF			MAXPAGLOCK	00000915	R	03
\$\$T1	= 00000000			MAXPASSCNT	00000020	R	02
\$\$T2	= 00000004			MEMLOOPCTL	00000568	R	03
ADJWSLERR	00000538	R	03	MEMLOOPCTLADR	000003BC	R	03
ADJWSLERRADR	0000027A	R	03	MEMLOOPCTLSIZ	= 0000007F		
ADJWSLERRSIZ	= 00000068			MSGBUF	000000FE	R	02
ADJWSLSUBR	00000865	R	03	MSGBUFD	000000E0	R	02
BIT	= 00000004			MSGBUFID	000000F0	R	02
CHECK1	00000719	R	03	MSGBUFSIZ	= 000000A0		
CHECK2	00000796	R	03	MSGLEN	000000DC	R	02
CNTREGERR	00000518	R	03	NOREADERR	00000550	R	03
CNTREGERRADR	00000196	R	03	NOREADERRADR	00000351	R	03
CNTREGERRSIZ	= 00000072			NOREADERRSIZ	= 00000024		
CNTREGSUBR	0000076C	R	03	NOWRITERR	00000560	R	03
CRETVAERR	00000508	R	03	NOWRITERRADR	00000397	R	03
CRETVAERRADR	0000000A	R	03	NOWRITERRSIZ	= 00000025		
CRETVAERRSIZ	= 00000063			OUTNAMADR	00000000	R	03
CRETVASUBR	000006CB	R	03	OUTNAMSIZ	= 0000000A		
CRLF	000000F0	R	02	PASSCNT	00000024	R	02
CTLSM_MEMLOOP	= 0000C001			PID	0000001C	R	02
CTLSM_PIDMSG	= 00000004			PIDCTL	00000588	R	03
CTLSM_RNGCHK	= 00000008			PIDCTLADR	00000505	R	03
CTLSM_TSTLOOP	= 00000002			PIDCTLSIZ	= 00000003		
CTLSV_MEMLOOP	= 00000000			PIDMSG	000000FA	R	02
CTLSV_PIDMSG	= 00000002			PIDMSGD	000000E8	R	02
CTLSV_RNGCHK	= 00000003			PREVPROT	00000048	R	02
CTLSV_TSTLOOP	= 00000001			PROBERR	000008F2	R	03
CTLFLG	00000010	R	02	PRT\$C_NONE	= 00000010		
DELTVAERR	00000510	R	03	RAB	00000098	R	02
DELTVAERRADR	0000006D	R	03	RAB\$B_RAC	= 0000001E		
DELTVAERRSIZ	= 00000063			RAB\$C_BID	= 00000001		
DELTVASUBR	000006DF	R	03	RAB\$C_BLN	= 00000044		
EXPREGERR	00000520	R	03	RAB\$C_SEQ	= 00000000		
EXPREGERRADR	00000208	R	03	RAB\$C_CTX	= 00000018		
EXPREGERRSIZ	= 00000072			RAB\$C_RBF	= 00000028		
EXPREGSUBR	00000782	R	03	RAB\$C_ROP	= 00000004		
FAB	00000048	R	02	RAB\$W_RSZ	= 00000022		
FAB\$C_BID	= 00000003			RANGECHK	000007EB	R	03
FAB\$C_BLN	= 00000050			RANGERR	00000570	R	03
FAB\$C_SEQ	= 00000000			RANGERRADR	0000043B	R	03
FAB\$C_VAR	= 00000002			RANGERRSIZ	= 0000004F		
FAB\$C_ALQ	= 00000010			READERR	00000548	R	03
FAB\$C_FOP	= 00000004			READERRADR	00000330	R	03
FAB\$C_CHAN_MODE	= 00000002			READERRSIZ	= 00000021		
FAB\$C_FILE_MODE	= 00000004			RETRANGE	00000008	R	02
FAB\$C_LNM_MODE	= 00000000			RSTART	0000060A	R	03
FAB\$C_PUT	= 00000000			RUN1_MSG	00000580	R	03
FAB\$C_GBC	= 00000048			RUN1_MSGADR	000004C3	R	03
HIGHPOADR	00000018	R	02	RUN1_MSGSIZ	= 00000042		
IDMSG	00000578	R	03	SAVEAD	00000014	R	02
IDMSGADR	0000048A	R	03	SIZ...	= 00000001		
IDMSGSIZ	= 00000039			SS\$_EXQUOTA	= 0000001C		
INRANGE	00000000	R	02	SS\$_INSFWSL	= 0000011C		


```

SS$ _LKWSETFUL = 00000194
SS$ _NORMAL    = 00000001
SS$ _VASFULL   = 00000244
SS$ _WASSET    = 00000009
START          = 00000590 R      03
SY$SADJWSL    ***** GX 03
SY$SCNTREG    ***** GX 03
SY$SCONNECT   ***** GX 03
SY$SCRETVA    ***** GX 03
SY$SDELTVA    ***** GX 03
SY$SEXIT      ***** GX 03
SY$SEXPREG    ***** GX 03
SY$SFAO       ***** X 03
SY$SLKWSET    ***** GX 03
SY$SOPEN      ***** GX 03
SY$SPUT       ***** GX 03
SY$SRESUME    ***** GX 03
SY$SULWSET    ***** GX 03
TYPEMSGBUF    00000888 R      03
ULWSETERR     00000530 R      03
ULWSETERRADR  00000133 R      03
ULWSETERRSIZ = 00000063
ULWSETSUBR    00000707 R      03
WRITERR       00000558 R      03
WRITERRADR    00000375 R      03
WRITERRSIZ    = 00000022
WRKSETDEF     0000002C R      02
WRKSETLIM     00000028 R      02
WRKSETMAX     = 00000034 R      02
WRKSETMAXADD  00000044 R      02
WRKSETMIN     = 0000003C R      02
WSETLMCTL     00000540 R      03
WSETLMCTLADR  000002E2 R      03
WSETLMCTLSIZ = 0000004E
    
```

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$AB\$\$	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
DATA0	0000019E (414.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC PAGE
CODE	00000958 (2392.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC PAGE

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	11	00:00:00.08	00:00:01.29
Command processing	93	00:00:00.80	00:00:05.15
Pass 1	326	00:00:11.60	00:00:38.06
Symbol table sort	0	00:00:01.28	00:00:03.19
Pass 2	151	00:00:02.84	00:00:10.16

Symbol table output	18	00:00:00.12	00:00:01.50
Psect synopsis output	2	00:00:00.01	00:00:00.49
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	603	00:00:16.75	00:00:59.98

The working set limit was 1500 pages.
65554 bytes (129 pages) of virtual memory were used to buffer the intermediate code.
There were 50 pages of symbol table space allocated to hold 900 non-local and 18 local symbols.
738 source lines were read in Pass 1, producing 21 object records in Pass 2.
54 pages of virtual memory were used to define 47 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

1151 GETS were required to define 30 macros.

There were no errors, warnings or information messages.

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

