

MMM	MMM	AAAAAAAAA	IIIIIIII	LLL
MMM	MMM	AAAAAAAAA	IIIIIIII	LLL
MMM	MMM	AAAAAAAAA	IIIIIIII	LLL
MMMMMM	MMMMMM	AAA	III	LLL
MMMMMM	MMMMMM	AAA	III	LLL
MMMMMM	MMMMMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAAAAAAAAAAA	III	LLL
MMM	MMM	AAAAAAAAAAAA	III	LLL
MMM	MMM	AAAAAAAAAAAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	III	LLL
MMM	MMM	AAA	IIIIIIII	LLLLLLLLLLLLLLLL
MMM	MMM	AAA	IIIIIIII	LLLLLLLLLLLLLLLL
MMM	MMM	AAA	IIIIIIII	LLLLLLLLLLLLLLLL

1
S
I
M
P
L
E
S
T
I
M
A
T
I
O
N
S
A
R
E
A
V
A
I
L
A
B
L
E
O
N
L
Y
I
N
T
H
E
S
E
P
A
G
E
S
O
F
T
H
E
B
O
O
K
A
R
E
A
V
A
I
L
A
B
L
E
O
N
L
Y
I
N
T
H
E
S
E
P
A
G
E
S
O
F
T
H
E
B
O
O
K

```

SSSSSSSS  EEEEEEEEE  NN      NN  DDDDDDDD  MM      MM  SSSSSSSS  GGGGGGGG
SSSSSSSS  EEEEEEEEE  NN      NN  DDDDDDDD  MM      MM  SSSSSSSS  GGGGGGGG
SS        EE        NN      NN  DD      DD  MMMM  MMMM  SS        GG
SS        EE        NN      NN  DD      DD  MMMM  MMMM  SS        GG
SS        EE        NNNN    NN  DD      DD  MM  MM  MM  SS        GG
SS        EE        NNNN    NN  DD      DD  MM  MM  MM  SS        GG
        SSSSSS  EEEEEEEEE  NN  NN  NN  DD      DD  MM      MM  SSSSSS  GG
        SSSSSS  EEEEEEEEE  NN  NN  NN  DD      DD  MM      MM  SSSSSS  GG
        SS      EE        NN      NNNN  DD      DD  MM      MM  SS      GG
        SS      EE        NN      NNNN  DD      DD  MM      MM  SS      GG
        SS      EE        NN      NN      DD      DD  MM      MM  SS      GG
        SS      EE        NN      NN      DD      DD  MM      MM  SS      GG
SSSSSSSS  EEEEEEEEE  NN      NN  DD      DD  MM      MM  SSSSSSSS  GG
SSSSSSSS  EEEEEEEEE  NN      NN  DD      DD  MM      MM  SSSSSSSS  GG

```

```

LL        IIIIII  SSSSSSSS
LL        IIIIII  SSSSSSSS
LL        II      SS
LL        II      SS
LL        II      SS
LL        II      SS
LL        II      SSSSSS
LL        II      SSSSSS
LL        II      SS
LL        II      SS
LL        II      SS
LL        II      SS
LLLLLLLLLL IIIIII  SSSSSSSS
LLLLLLLLLL IIIIII  SSSSSSSS

```



```

1 0001 0 MODULE MAIL$SEND_MSG( %TITLE 'VAX/VMS PERSONAL MAIL UTILITY'
2 0002 0 IDENT = 'V04-000' ) =
3 0003 1 BEGIN
4 0004 1
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
10 0010 1 * ALL RIGHTS RESERVED. *
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
17 0017 1 * TRANSFERRED. *
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
21 0021 1 * CORPORATION. *
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
25 0025 1 *
26 0026 1 *****
27 0027 1
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: VAX/VMS MAIL UTILITY
32 0032 1
33 0033 1 ABSTRACT: Send message to another user on local node
34 0034 1
35 0035 1 ENVIRONMENT: NATIVE/USER MODE
36 0036 1
37 0037 1 AUTHOR: Benn Schreiber, CREATION DATE: 22-Dec-1982
38 0038 1
39 0039 1 MODIFIED BY:
40 0040 1
41 0041 1 V03-019 ROP0032 Robert Posniak 09-AUG-1984
42 0042 1 Make sure locally allocated XABs are disconnected
43 0043 1 from global FABs.
44 0044 1
45 0045 1 V03-018 ROP0029 Robert Posniak 24-JUL-1984
46 0046 1 Allow VFC format files to be sent in
47 0047 1 block mode.
48 0048 1
49 0049 1 V03-017 ROP0013 Robert Posniak 27-JUN-1984
50 0050 1 Only send in block mode if input file has
51 0051 1 variable length records. Modify test of
52 0052 1 isam_create status to allow sending to
53 0053 1 sequential files.
54 0054 1
55 0055 1 V03-016 BLS0272 Benn Schreiber 18-FEB-1984 13:39:01
56 0056 1 Change call to mail$net_send to send addressee descriptor
57 0057 1

```

58	0058	1	V03-015	BLS0263	Benn Schreiber	5-FEB-1984
59	0059	1			Only set exec mode on external file if requested.	
60	0060	1				
61	0061	1	V03-014	BLS0255	Benn Schreiber	28-Dec-1983
62	0062	1			New global flags. Open mail file with exec mode logicals.	
63	0063	1			Fix error return from create isam to only try sequential	
64	0064	1			file if error was file locked. Skip addressees marked nosend.	
65	0065	1				
66	0066	1	V03-013	BLS0250	Benn Schreiber	9-Dec-1983
67	0067	1			Ensure that we are writing to a disk device.	
68	0068	1				
69	0069	1	V03-012	BLS0246	Benn Schreiber	8-Nov-1983
70	0070	1			New mail count in VMSMAIL.DAT, rather than mail file	
71	0071	1				
72	0072	1	V03-011	BLS0235	Benn Schreiber	23-Aug-1983
73	0073	1			Quit if \$connect fails, also. Really rewind the	
74	0074	1			message file for each addressee.	
75	0075	1				
76	0076	1	V03-010	BLS0233	Benn Schreiber	7-Aug-1983
77	0077	1			Put nodename on sender's name correctly	
78	0078	1				
79	0079	1	V03-009	BLS0229	Benn Schreiber	16-Jul-1983
80	0080	1			Deliver external message files in block mode. Rewrite	
81	0081	1			mail\$send_addr into bliss.	
82	0082	1				
83	0083	1	V03-008	BLS0222	Benn Schreiber	15-May-1983
84	0084	1			Set SQO on sequential external file.	
85	0085	1				
86	0086	1	V03-007	BLS0221	Benn Schreiber	5-May-1983
87	0087	1			Correct SYSPRV handling	
88	0088	1				
89	0089	1	V03-006	BLS0220	Benn Schreiber	30-Apr-1983
90	0090	1			Device/directory are now one argument	
91	0091	1				
92	0092	1	V03-005	BLS0218	Benn Schreiber	18-Apr-1983
93	0093	1			Call check_date with 3rd arg (dev/dir descriptor). Allow	
94	0094	1			for no protection xab and make one up.	
95	0095	1				
96	0096	1	V03-004	BLS0217	Benn Schreiber	12-Apr-1983
97	0097	1			Tell if new folder created. Add external message deliver	
98	0098	1			for large messages.	
99	0099	1				
100	0100	1	V03-003	BLS0216	Benn Schreiber	1-Apr-1983
101	0101	1			Add flag to use passed in date/time rather than current.	
102	0102	1				
103	0103	1	V03-002	BLS0214	Benn Schreiber	27-Mar-1983
104	0104	1			Put new messages in NEWMAIL folder. Count records	
105	0105	1			and store count in message header.	
106	0106	1				

```
108 0107 1 |
109 0108 1 | TABLE OF CONTENTS:
110 0109 1 |
111 0110 1 |
112 0111 1 |
113 0112 1 | INCLUDE FILES:
114 0113 1 |
115 0114 1 | LIBRARY 'SYSSLIBRARY:LIB';
116 0115 1 | REQUIRE 'SRCS:MAILREQ';
117 0261 1 | LIBRARY 'LIBS:MAILDEF';
118 0262 1 |
119 0263 1 | EXTERNAL
120 0264 1 | MAIL$GL_BIGBUFSIZ, | Size of BIGBUF
121 0265 1 | MAIL$GL_BIGBUFFER, | 0 or pointer to bigbufsiz pgs
122 0266 1 | MAIL$GL_FLAGS : $BBLOCK, | Control flags
123 0267 1 | MAIL$Q_SYSNODE2 : $BBLOCK, | 'node:;'
124 0268 1 | MAIL$Q_FULLUSER, | Full user name
125 0269 1 | MAIL$T_UAF_BUF : REF $BBLOCK, | SYSuaf buffer
126 0270 1 | MAIL$SL_MYUIC, | UIC of person running MAIL
127 0271 1 | MAIL$SL_SIZLIM, | Max size msg to put in isam
128 0272 1 | MAIL$SD_EXTMSGNAM, | FAO string to make file name
129 0273 1 | MAIL$SD_NEWMAIL; | Addr of descr of 'NEWMAIL'
130 0274 1 |
131 0275 1 | EXTERNAL ROUTINE
132 0276 1 | LIB$FREE_VM, | Deallocate memory
133 0277 1 | MAIL$CHECK_DATE, | See if key in ISAM file
134 0278 1 | MAIL$CREATE_ISAM, | Create ISAM file
135 0279 1 | MAIL$DELETE_LINK, | Delete all logical links
136 0280 1 | MAIL$FAO_GETMSG, | Get message text and fao it
137 0281 1 | MAIL$GET_DEVDIR, | Get user's mail device/dir
138 0282 1 | MAIL$GETMSG, | Get message text for code
139 0283 1 | MAIL$INCR_NEWMAIL, | Increment new mail count
140 0284 1 | MAIL$NET_SEND, | Send to remote addressee
141 0285 1 | MAIL$NOTIFY, | Notify message received
142 0286 1 | MAIL$UP_PRIV, | Raise privileges
143 0287 1 | MAIL$DOWN_PRIV, | Drop privileges
144 0288 1 | STR$CONCAT, | String concatenation
145 0289 1 | STR$COPY_DX, | String copy
146 0290 1 | SYSS$FAO, | Formatted ascii output
147 0291 1 | UTIL$GETFILENAME, | Ditto
148 0292 1 | UTIL$REPORT_IO_ERROR; | Report I/O error
149 0293 1 |
150 0294 1 | OWN
151 0295 1 | DYN_DESC : DYN_DESC_DECL, | Two dynamic strings
152 0296 1 | DYN_DESC_2 : DYN_DESC_DECL;
153 0297 1 |
154 P 0298 1 | $SHR_MSGDEF(MAIL,126,LOCAL,
155 P 0299 1 | (OPENOUT,ERROR),
156 P 0300 1 | (READERR,ERROR),
157 P 0301 1 | (WRITEERR,WARNING),
158 0302 1 | (CLOSEOUT,WARNING));
159 0303 1 |
160 0304 1 | MACRO
161 0305 1 | DEFFILENAM = 'MAIL'%, | Default message file name
162 0306 1 | DEFFILETYP = '.MAI'%; | Default message file type
163 0307 1 |
164 0308 1 | LITERAL
```

```
: 165      0309 1      PRO_SYS      = $BITPOSITION(XABS$V_SYS),  
: 166      0310 1      PRO_OWN      = $BITPOSITION(XABS$V_OWN),  
: 167      0311 1      PRO_GRP      = $BITPOSITION(XABS$V_GRP),  
: 168      0312 1      PRO_WLD      = $BITPOSITION(XABS$V_WLD),  
: 169      0313 1  
: 170      0314 1      SND_PRO      =  
: 171      0315 1 (      XABS$M_NOEXE OR XABS$M_NODEL) ^ PRO_SYS OR  
: 172      0316 1 (      XABS$M_NOEXE OR XABS$M_NODEL) ^ PRO_OWN OR  
: 173      0317 1 (XABS$M_NOREAD OR XABS$M_NOWRITE OR XABS$M_NOEXE OR XABS$M_NODEL) ^ PRO_GRP OR  
: 174      0318 1 (XABS$M_NOREAD OR XABS$M_NOWRITE OR XABS$M_NOEXE OR XABS$M_NODEL) ^ PRO_WLD;
```

```

176 0319 1 ROUTINE PUT_SEND_MSG (LENGTH, ADDR, SND_RAB) =
177 0320 2 BEGIN
178 0321 2 ++
179 0322 2 FUNCTIONAL DESCRIPTION:
180 0323 2
181 0324 2 This routine write the record specified by LENGTH and ADDR
182 0325 2 to SND_RAB, and reports errors encountered.
183 0326 2
184 0327 2 INPUTS:
185 0328 2
186 0329 2 LENGTH length of record to write
187 0330 2 ADDR address of record to write
188 0331 2
189 0332 2 --
190 0333 2 MAP
191 0334 2 SND_RAB : REF $BBLOCK;
192 0335 2
193 0336 2 LOCAL
194 0337 2 STATUS;
195 0338 2
196 0339 2 SND_RAB[RAB$W_RSZ] = .LENGTH;
197 0340 2 SND_RAB[RAB$L_RBF] = .ADDR;
198 0341 2
199 P 0342 2 IF_ERR((IF .SND_RAB[RAB$V_BIO]
200 P 0343 2 THEN $WRITE(RAB=.SND_RAB,ERR=UTIL$REPORT_ID_ERROR)
201 P 0344 2 ELSE $PUT(RAB=.SND_RAB,ERR=UTIL$REPORT_ID_ERROR));,
202 0345 2 RETURN .STATUS);
203 0346 2 RETURN TRUE
204 0347 1 END;

```

```

.TITLE MAIL$SEND_MSG VAX/VMS PERSONAL MAIL UTILITY
.IDENT \V04-000\
.PSECT $OWNS,NOEXE,2

```

```

00# 0000 DYN_DESC:
02 0E 00002 .BYTE 0[2]
00004 .BYTE 14, 2
00008 DYN_DESC_2:
02 0E 0000A .BLKB 4
0000C .BLKB 4

```

```

.EXTRN MAIL$GL_BIGBUFSIZ
.EXTRN MAIL$GL_BIGBUFFER
.EXTRN MAIL$GL_FLAGS, MAIL$Q_SYSNODE2
.EXTRN MAIL$Q_FULLUSER
.EXTRN MAIL$T_UAF_BUFF
.EXTRN MAIL$L_MYUTC, MAIL$L_SIZLIM
.EXTRN MAIL$SD_EXTMSGNAM
.EXTRN MAIL$SD_NEWMAIL
.EXTRN LIB$FREE_VM, MAIL$CHECK_DATE
.EXTRN MAIL$CREATE_ISAM
.EXTRN MAIL$DELETECINK
.EXTRN MAIL$FAO_GETMSG

```

```
.EXTRN MAIL$GET_DEVDIR
.EXTRN MAIL$GETMSG, MAIL$INCR_NEWMAIL
.EXTRN MAIL$NET_SEND, MAIL$NOTIFY
.EXTRN MAIL$UP_PRIV, MAIL$DOWN_PRIV
.EXTRN STR$CONCAT, STR$COPY_DX
.EXTRN SYSS$FAD, UTIL$GETFILENAME
.EXTRN UTIL$REPORT_IO_ERROR
.EXTRN SYSS$WRITE, SYSS$PUT
```

```
.PSECT $CODE$,NOWRT,2
```

```
0004 00000 PUT_SEND_MSG:
:WORD Save R2 : 0319
MOVAB UTIL$REPORT_IO_ERROR, R2 : 0339
MOVL SND_RAB, R0 : 0340
MOVW LENGTH, 34(R0) : 0345
MOVL ADDR, 40(R0)
BBC #3, 5(R0), 1$
PUSHR #*M<R0,R2>
CALLS #2, SYSS$WRITE
BRB 2$
PUSHR #*M<R0,R2>
CALLS #2, SYSS$PUT
BLBC STATUS, 3$ : 0346
MOVL #1, R0 : 0347
RET
```

```
; Routine Size: 55 bytes, Routine Base: $CODE$ + 0000
```



```

: 206 0348 1 ROUTINE PUT_NEW_REC(LENGTH,ADDR,SND_RAB) =
: 207 0349 1 |++
: 208 0350 1 | FUNCTIONAL DESCRIPTION:
: 209 0351 1 |
: 210 0352 1 |         Write a record to the isam file.
: 211 0353 1 |
: 212 0354 1 | --
: 213 0355 2 BEGIN
: 214 0356 2
: 215 0357 2 MAP
: 216 0358 2     SND_RAB : REF $BBLOCK,
: 217 0359 2     ADDR : REF $BBLOCK;
: 218 0360 2
: 219 0361 2 BUILTIN
: 220 0362 2     CALLG,
: 221 0363 2     AP;
: 222 0364 2
: 223 0365 2 SND_RAB[RAB$L_KBF] = ADDR[MFR_Q_DATIM];
: 224 0366 2 SND_RAB[RAB$B_KSZ] = 8;
: 225 0367 2
: 226 0368 2 RETURN CALLG(.AP,PUT_SEND_MSG)
: 227 0369 2
: 228 0370 1 END;

```

```

                                0000 0000 PUT_NEW_REC:
                                .WORD   Save nothing           : 0348
                                MOVL    SND_RAB, R0             : 0365
                                MOVL    ADDR, 48(R0)           :
                                MOVB    #8, 52(R0)             : 0366
                                CALLG   (AP), PUT_SEND_MSG     : 0368
                                RET

```

: Routine Size: 20 bytes, Routine Base: \$CODE\$ + 0037

```

: 230 0371 1 ROUTINE END_PACK(RECLEN,RECBUF,SND_RAB) =
: 231 0372 1 --
: 232 0373 1 FUNCTIONAL DESCRIPTION:
: 233 0374 1
: 234 0375 1 Write out remainder of packed record
: 235 0376 1
: 236 0377 1 --
: 237 0378 2 BEGIN
: 238 0379 2
: 239 0380 2 BUILTIN
: 240 0381 2 CALLG,
: 241 0382 2 AP;
: 242 0383 2
: 243 0384 2 IF .RECLEN EQL $BYTEOFFSET(MFR_T_DATA)
: 244 0385 2 THEN RETURN TRUE;
: 245 0386 2
: 246 0387 2 RETURN CALLG(.AP,PUT_NEW_REC)
: 247 0388 2
: 248 0389 1 END;

```

```

                                0000 00000 END_PACK:
                                .WORD Save nothing
                                30      04 AC D1 00002      CMPL RECLEN, #48      : 0371
                                04      12 00006      BNEQ 1$      : 0384
                                50      01 D0 00008      MOVL #1, R0      : 0385
                                DC AF      6C FA 0000C 1$: RET      : 0387
                                04      00010      RET      : 0389

```

: Routine Size: 17 bytes, Routine Base: \$CODE\$ + 004B

```

: 250      0390 1 ROUTINE INIT_PACK_REC(RECLEN) =
: 251      0391 1 |--
: 252      0392 1 | FUNCTIONAL DESCRIPTION:
: 253      0393 1 |
: 254      0394 1 |       Initialize for packing records
: 255      0395 1 |
: 256      0396 1 |--
: 257      0397 2 BEGIN
: 258      0398 2
: 259      0399 2 MAP
: 260      0400 2     RECLEN : REF VECTOR[,LONG];
: 261      0401 2
: 262      0402 2     RECLEN[0] = $BYTEOFFSET(MFR_T_DATA);
: 263      0403 2
: 264      0404 2     RETURN TRUE
: 265      0405 1     END;

```

```

                                0000 00000 INIT_PACK_REC:
                                .WORD      Save nothing
                                04  BC      30  D0 00002      MOVL      #48, @RECLEN      : 0390
                                01  D0 00006      MOVL      #1, R0          : 0402
                                04 00009      RET              : 0404
:                               : 0405

```

; Routine Size: 10 bytes, Routine Base: \$CODE\$ + 005C

```

: 267 0406 1 ROUTINE PACK_NEW_REC (LENGTH,ADDR,RECLN,RECBUF,SND_RAB) =
: 268 0407 1 |--
: 269 0408 1 | FUNCTIONAL DESCRIPTION:
: 270 0409 1 |
: 271 0410 1 |         Pack records into buffer.  Write out records as needed
: 272 0411 1 |
: 273 0412 1 | |--
: 274 0413 2 BEGIN
: 275 0414 2
: 276 0415 2 MAP
: 277 0416 2     RECLN : REF VECTOR[,LONG];
: 278 0417 2
: 279 0418 2 IF (.RECLN[0] + .LENGTH + 2) GTR MFR_C_MAXLEN
: 280 0419 3 THEN BEGIN
: 281 P 0420 3     IF_ERR(PUT_NEW_REC(.RECLN[0],.RECBUF,.SND_RAB));,
: 282 0421 3         RETURN .STATUS);
: 283 0422 3     INIT_PACK_REC(RECLN[0]);
: 284 0423 2     END;
: 285 0424 2
: 286 0425 3 BEGIN
: 287 0426 3     BIND
: 288 0427 3         BUFF = .RECBUF + .RECLN[0] : VECTOR[,WORD];
: 289 0428 3
: 290 0429 3         BUFF[0] = .LENGTH;
: 291 0430 3         CH$MOVE(.LENGTH,.ADDR,BUFF[1]);
: 292 0431 3         RECLN[0] = .RECLN[0] + .LENGTH + 2;
: 293 0432 2     END;
: 294 0433 2
: 295 0434 2 RETURN TRUE
: 296 0435 1 END;

```

```

                                003C 00000 PACK_NEW_REC:
                                :WORD
                                Save R2,R3,R4,R5
04  AC  C1 00002          ADDL3 LENGTH, @RECLN, R0      : 0406
02  C0 00008          ADDL2 #2, R0      : 0418
00000800 8F          CMPL R0, #2048
15  15 00012          BLEQ 1$
10  AC  7D 00014          MOVQ  RECBUF, -(SP)      : 0421
0C  BC  DD 00018          PUSHL @RECLN
03  FB 0001B          CALLS #3, PUT_NEW_REC
50  E9 0001F          BLBC STATUS, -2$
0C  AC  DD 00022          PUSHL RECLN      : 0422
01  FB 00025          CALLS #1, INIT_PACK_REC
50  C1 00029 1$:      ADDL3 @RECLN, -RECBUF, R0      : 0427
04  AC  B0 0002F          MOVW  LENGTH, (R0)      : 0429
02  A0  08 BC 00033          MOVW  LENGTH, @ADDR, 2(R0)      : 0430
50  0C  BC 0003A          ADDL3 LENGTH, @RECLN, R0      : 0431
0C  BC 00040          MOVAB 2(R0), @RECLN
02  A0  9E 00040          MOVAB #1, R0      : 0434
01  D0 00045          MOVL  #1, R0
04  00048 2$:      RET      : 0435

```

: Routine Size: 73 bytes, Routine Base: \$CODES + 0066

MAIL\$SEND_MSG VAX/VMS PERSONAL MAIL UTILITY
V04-000

D 6
16-Sep-1984 01:17:01
14-Sep-1984 12:42:31

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[MAIL.SRC]SENDMSG.B32;1 Page 11
(7)

MA
VO

```

: 298 0436 1 ROUTINE GET_RECORD (OUTDESC,GETRAB) =
: 299 0437 1 |++
: 300 0438 1 | FUNCTIONAL DESCRIPTION:
: 301 0439 1 |
: 302 0440 1 |         Read a record from the temp file
: 303 0441 1 |
: 304 0442 1 | --
: 305 0443 2 BEGIN
: 306 0444 2
: 307 0445 2 MAP
: 308 0446 2     OUTDESC : REF VECTOR[,LONG],
: 309 0447 2     GETRAB  : REF $BLOCK;
: 310 0448 2
: 311 0449 2 LOCAL
: 312 0450 2     STATUS;
: 313 0451 2
: 314 0452 3 STATUS = (IF .GETRAB[RAB$V_BIO]
: 315 0453 4     THEN $READ(RAB=.GETRAB)
: 316 0454 2     ELSE $GET(RAB=.GETRAB));
: 317 0455 2 IF .STATUS
: 318 0456 3 THEN BEGIN
: 319 0457 3     OUTDESC[0] = .GETRAB[RAB$W_RSZ];
: 320 0458 3     OUTDESC[1] = .GETRAB[RAB$L_RBF];
: 321 0459 2     END;
: 322 0460 2
: 323 0461 2 RETURN .STATUS
: 324 0462 1 END;

```

.EXTRN SYS\$READ, SYS\$GET

```

                                0004 0000 GET_RECORD:
                                .WORD  Save R2
0B      05      52      08      AC  DO 00002      MOVL  GETRAB, R2      : 0436
                                03  E1 00006      BBC   #3, 5(R2), 1$   : 0452
                                52  DD 0000B      PUSHL R2
00000000G  00      01  FB 0000D      CALLS #1, SYS$READ      : 0453
                                09  11 00014      BRB   2$
                                52  DD 00016 1$:    PUSHL R2
00000000G  00      01  FB 00018      CALLS #1, SYS$GET      : 0454
                                50  E9 0001F 2$:    BLBC  STATUS, 3$
                                51      04      AC  DO 00022      MOVL  OUTDESC, R1      : 0455
                                22      A2      3C 00026      MOVZWL 34(R2), (R1)      : 0457
                                04      61      28      A2      DO 0002A      MOVL  40(R2), 4(R1)      : 0458
                                04  0002F 3$:    RET

```

: Routine Size: 48 bytes, Routine Base: \$CODE\$ + 00AF

```
326 0463 1 GLOBAL ROUTINE MAIL$WRITE_MSG
327 0464 1 (GET_ROUTINE,CNCTDESC,FOLDER_DESC,MSGFLAGS,OUTRAB,DATETIME) =
328 0465 1 !++
329 0466 1 ! FUNCTIONAL DESCRIPTION:
330 0467 1 !
331 0468 1 ! Write a message to an ISAM mail file
332 0469 1 !
333 0470 1 ! --
334 0471 2 BEGIN
335 0472 2
336 0473 2 MAP
337 0474 2 CNCTDESC : REF $BBLOCK,
338 0475 2 FOLDER_DESC : REF $BBLOCK,
339 0476 2 MSGFLAGS : $BBLOCK,
340 0477 2 OUTRAB : REF $BBLOCK,
341 0478 2 DATETIME : REF VECTOR[,LONG];
342 0479 2
343 0480 2 BIND
344 0481 2 GET_RAB = CNCTDESC[CNCT_T_RAB] : $BBLOCK,
345 0482 2 FROM_DESC = CNCTDESC[CNCT_Q_FROMDESC] : $BBLOCK,
346 0483 2 TO_DESC = CNCTDESC[CNCT_Q_TODESC] : $BBLOCK,
347 0484 2 SUBJ_DESC = CNCTDESC[CNCT_Q_SUBJECT] : $BBLOCK;
348 0485 2
349 0486 2 LOCAL
350 0487 2 PTR : REF VECTOR[WORD],
351 0488 2 PTR1 : REF VECTOR[,LONG],
352 0489 2 MSGHDR : REF $BBLOCK,
353 0490 2 SAVERFA : VECTOR[2,LONG],
354 0491 2 RECLN,
355 0492 2 TFAB : REF $BBLOCK,
356 0493 2 SIZEOFF : REF VECTOR[,LONG],
357 0494 2 RECOUNT,
358 0495 2 MMHLEN,
359 0496 2 EXTMSG,
360 0497 2 DEVDIRDESC : VECTOR[2,LONG],
361 0498 2 OUTNAM : REF $BBLOCK,
362 0499 2 MMHRFA : VECTOR[2,LONG],
363 0500 2 RECDISC : VECTOR[2,LONG],
364 0501 2 TMPFAB : $FAB_DECL,
365 0502 2 TMPRAB : $RAB_DECL,
366 0503 2 TMPNAM : $NAM_DECL,
367 0504 2 TMPXABPRO : $XABPRO_DECL,
368 0505 2 NAMBUF : $BBLOCK[NAM$C_MAXRSS],
369 0506 2 SAVEMMH : $BBLOCK[MFR_C_MAXLEN],
370 0507 2 PROXAB : REF $BBLOCK,
371 0508 2 MMH : $BBLOCK[MFR_C_MAXLEN];
372 0509 2
373 0510 2 OUTRAB[RAB$S_UBF] = MMH;
374 0511 2 OUTRAB[RAB$S_USZ] = MFR_C_MAXLEN;
375 0512 2 OUTRAB[RAB$S_RAC] = RAB$C_KEY;
376 0513 2 OUTRAB[RAB$S_ROP] = 0;
377 0514 2 OUTRAB[RAB$S_CTX] = MAIL$WRITEERR;
378 0515 2
379 0516 2 ! Create the mail message header record
380 0517 2
381 0518 2 IF .MSGFLAGS[MCR_V_USE_DATIM] NEQ 0
382 0519 2 THEN CH$MOVE(9,.DATETIME,MMH[MFR_Q_DATIM])
```

```
383 0520 3 ELSE BEGIN
384 0521 3   $GETTIM(TIMADR=MMH[MFR_Q_DATIM]);
385 0522 3   IF NOT MAIL$CHECK_DATE(.OUTRAB,MMH[MFR_Q_DATIM])
386 0523 4   THEN BEGIN
387 0524 4     SIGNAL(MAIL$_DATIMUSED,2,MMH[MFR_Q_DATIM],UTIL$GETFILENAME(.OUTRAB));
388 0525 4     RETURN MAIL$_DATIMUSED;
389 0526 3   END;
390 0527 2   END;
391 0528 2 MSGHDR = MMH[MFR_T_DATA];
392 0529 2 PTR = MSGHDR[MMH_Q_DATID];
393 0530 2 (.PTR)<0,32,0> = (.MMH[MFR_Q_DATIM])<16,32,0>;
394 0531 2 (.PTR)<32,16,0> = (.MMH[MFR_Q_DATIM])<48,16,0>;
395 0532 2 (.PTR)<48,16,0> = 4;
396 0533 2
397 0534 2 : If message is gtr MAIL$_SIZLIM blocks, then deliver message to
398 0535 2 : a sequential file.
399 0536 2
400 0537 2 EXTMSG = 0;
401 0538 2 IF .CNCTDESC[CNCT_L_MSGBLKS] GTRU .MAIL$_SIZLIM
402 0539 2 THEN BEGIN
403 0540 3   EXTMSG = 1;
404 0541 3   OUTNAM = .OUTRAB[RAB$_FAB];
405 0542 3   PROXAB = .OUTNAM[FAB$_XAB]; !Protection XAB first in chain
406 0543 3   : Zero XAB happens only if COPY command into current file.
407 0544 3   IF .PROXAB EQL 0
408 0545 4   THEN BEGIN
409 P 0546 4     $XABPRO_INIT(XAB=TMPXABPRO,
410 0547 4     _NXT=0);
411 0548 4     TMPXABPRO[XAB$_UIC] = .MAIL$_MYUIC;
412 0549 4     PROXAB = TMPXABPRO;
413 0550 3   END;
414 0551 3   PROXAB[XAB$_PRO] = SND_PRO;
415 0552 3   PROXAB[XAB$_NXT] = 0;
416 0553 3   OUTNAM = .OUTNAM[FAB$_NAM];
417 0554 3   DEVDIRDESC[0] = .OUTNAM[NAM$_DEV] + .OUTNAM[NAM$_DIR];
418 0555 3   DEVDIRDESC[1] = .OUTNAM[NAM$_DEV];
419 0556 3   RECDESC[0] = NAM$_MAXRSS;
420 0557 3   RECDESC[1] = NAM$_BUF;
421 0558 3   IF NOT MAIL$CHECK_DATE(.OUTRAB,MSGHDR[MMH_Q_DATID],DEVDIRDESC)
422 0559 4   THEN BEGIN
423 0560 4     SIGNAL(MAIL$_DATIMUSED,2,MSGHDR[MMH_Q_DATID],
424 0561 4     UTIL$GETFILENAME(.OUTRAB));
425 0562 4     RETURN MAIL$_DATIMUSED;
426 0563 3   END;
427 0564 3   SYS$FAO(MAIL$_EXTMSGNAM,RECDESC,RECDESC,
428 0565 3     DEVDIRDESC,.(MSGHDR[MMH_Q_DATID])<32,32,0>,
429 0566 3     .(MSGHDR[MMH_Q_DATID])<0,32,0>);
430 P 0567 3   $FAB_INIT(FAB=TMPFAB,
431 P 0568 3     FNS=.RECDESC[0],
432 P 0569 3     FNA=.RECDESC[1],
433 P 0570 3     ALQ=.CNCTDESC[CNCT_L_MSGBLKS],
434 P 0571 3     CTX=MAIL$_OPENOUT,
435 P 0572 3     NAM=TMPNAM,
436 P 0573 3     XAB=.PROXAB,
437 P 0574 3     FAC=(PUT,TRN),
438 P 0575 3     FOP=SQO,
439 P 0576 3     RFM=VAR,
```



```
440 0577 3      RAT=(CR);
441 0578 3
442 0579 3      IF .MSGFLAGS[MCR V EXEC]
443 0580 3      THEN TMPFAB[FAB$V_LNM_MODE] = PSL$C_EXEC;
444 0581 3
445 0582 3      If rab in the cnct block is the message source, we can use block i/o
446 0583 3      to transfer the message.
447 0584 3
448 0585 3      TFAB = .GET_RAB[RAB$L_FAB];
449 0586 3
450 0587 3      Send in block mode only if input file has variable length records
451 0588 3      or VFC format.
452 0589 3
453 0590 3      IF (.MSGFLAGS[MCR V CNCT_RAB] NEQ 0) AND
454 0591 4      ((.TFAB[FAB$B_RFM] EQL FAB$C_VAR) OR (.TFAB[FAB$B_RFM] EQL FAB$C_VFC))
455 0592 4      THEN BEGIN
456 0593 4
457 0594 4      If file is open with fab$V_bro set, then we can enable block i/o.
458 0595 4      Copy file attributes, and disconnect/reconnect if not already in
459 0596 4      block mode
460 0597 4
461 0598 4      IF .TFAB[FAB$V_BRO]
462 0599 5      THEN BEGIN
463 0600 5      TMPFAB[FAB$B_RAT] = .TFAB[FAB$B_RAT];
464 0601 5      TMPFAB[FAB$B_RFM] = .TFAB[FAB$B_RFM];
465 0602 5      IF NOT .GET_RAB[RAB$V_BIO]
466 0603 6      THEN BEGIN
467 0604 6      $DISCONNECT(RAB=GET_RAB);
468 0605 6      GET_RAB[RAB$V_BIO] = TRUE;
469 0606 6      GET_RAB[RAB$L_UBF] = .MAIL$GL_BIGBUFFER;
470 0607 6      GET_RAB[RAB$W_USZ] = .MAIL$GL_BIGBUFSIZ*512;
471 P 0608 6      IF _ERR($CONNECT(RAB=GET_RAB,ERR=UTIL$REPORT_IO_ERROR));,
472 0609 6      RETURN .STATUS);
473 0610 5      END;
474 0611 4      END;
475 0612 3      END;
476 P 0613 3      $RAB_INIT(RAB=TMPRAB,
477 P 0614 3      FAB=TMPFAB,
478 0615 3      CTX=MAIL$_OPENOUT);
479 0616 3
480 0617 3      TMPFAB[FAB$V_BRO] = TMPRAB[RAB$V_BIO] = .GET_RAB[RAB$V_BIO];
481 0618 3
482 P 0619 3      $NAM_INIT(NAM=TMPNAM,
483 P 0620 3      ESS=NAM$C_MAXRSS,
484 P 0621 3      ESA=NAM$B,
485 P 0622 3      RSS=NAM$C_MAXRSS,
486 0623 3      RSA=NAM$B);
487 0624 3
488 P 0625 3      IF _ERR($CREATE(FAB=TMPFAB,ERR=UTIL$REPORT_IO_ERRGR));,
489 0626 3      RETURN .STATUS);
490 0627 3      IF NOT .TMPFAB[$FAB_DEV(DIR)]
491 0628 3      OR .TMPFAB[$FAB_DEV(FOR)]
492 0629 4      THEN BEGIN
493 0630 4      $CLOSE(FAB=TMPFAB);
494 0631 4      TMPFAB[FAB$L_STS] = SSS_IVDEVNAM;
495 0632 4      TMPFAB[FAB$L_STV] = 0;
496 0633 4      RETURN SIGNAL(SSS_IVDEVNAM)
```

```
497      0634      3      END;
498      0635      3
499      P 0636      3      IF_ERR($CONNECT(RAB=TMPRAB,ERR=UTIL$REPORT_IO_ERROR);,
500      P 0637      3      TMPFAB[FAB$V DLT] = TRUE;
501      P 0638      3      $CLOSE(FAB=TMPFAB);
502      0639      3      RETURN .STATUS);
503      0640      3      RECOUNT = 0;
504      0641      3      TMPRAB[RAB$L_CTX] = MAIL$WRITEERR;
505      0642      3      WHILE 1
506      0643      3      DO BEGIN
507      P 0644      4          IF_ERR((.GET_ROUTINE)(RECDESC,CNCTDESC[CNCT_T_RAB]);,
508      P 0645      4              IF .STATUS EQL RMSS_EOF
509      P 0646      4                  THEN EXITLOOP
510      P 0647      4                  ELSE BEGIN
511      P 0648      4                      UTIL$REPORT_IO_ERROR(CNCTDESC[CNCT_T_RAB]);
512      P 0649      4                      TMPFAB[FAB$V DLT] = TRUE;
513      P 0650      4                      $CLOSE(FAB=TMPFAB);
514      P 0651      4                      RETURN .STATUS;
515      0652      4                  END);
516      0653      4          IF NOT .TMPRAB[RAB$V BIO]
517      0654      4              AND .RECDESC[0] GEQU 2
518      0655      4              AND (.RECDESC[1] + .RECDESC[0] - 2) < 0,16 >
519      0656      5                  EQLU (CR OR LF^8)
520      0657      4              THEN RECDESC[0] = .RECDESC[0] - 2;
521      P 0658      4          IF_ERR(PUT_SEND_MSG(.RECDESC[0],.RECDESC[1],
522      P 0659      4              TMPRAB);,
523      P 0660      4              TMPFAB[FAB$V DLT] = TRUE;
524      P 0661      4              $CLOSE(FAB=TMPFAB);
525      0662      4              RETURN .STATUS);
526      0663      4          RECOUNT = .RECOUNT + 1;
527      0664      4      END;
528      0665      3
529      0666      3      ! If block mode copy and we don't know number of records from a previous
530      0667      3      ! delivery, then read the file in record mode and compute the # records
531      0668      3
532      0669      3      IF .TMPRAB[RAB$V BIO]
533      0670      3      THEN IF (RECOUNT = .CNCTDESC[CNCT_L_NRECS]) EQL 0
534      0671      4      THEN BEGIN
535      0672      4          $DISCONNECT(RAB=TMPRAB);
536      0673      4          TMPRAB[RAB$V BIO] = FALSE;
537      0674      4          TMPRAB[RAB$W_USZ] = MAIL$K INBUFFSZ;
538      0675      4          TMPRAB[RAB$L_UBF] = CNCTDESC[CNCT_T_BUFFER];
539      0676      5          IF $CONNECT(RAB=TMPRAB,ERR=UTIL$REPORT_IO_ERROR)
540      0677      5          THEN WHILE $GET(RAB=TMPRAB)
541      0678      4              DO RECOUNT = .RECOUNT + 1;
542      0679      3          END;
543      0680      3          TMPFAB[FAB$L_CTX] = MAIL$CLOSEOUT;
544      0681      3          $CLOSE(FAB=TMPFAB,ERR=UTIL$REPORT_IO_ERROR);
545      0682      3      END
546      0683      3
547      0684      3      ! Not too big, ensure date/time key not used
548      0685      3
549      0686      3      ELSE IF NOT MAIL$CHECK_DATE(.OUTRAB,MSGHDR[MMH_Q_DATID])
550      0687      3      THEN BEGIN
551      0688      3          SIGNAL 'MAIL$_DATIMUSED 2,MSGHDR[MMH_Q_DATID],
552      0689      3              UTIL$GETFILENAME(.OUTRAB));
553      0690      3      RETURN MAIL$_DATIMUSED;
```

```
554 0691 2 END;
555 0692 2
556 0693 2 Write the message date/id record to the file to 'reserve' it. Only
557 0694 2 one 8-byte record is written, which is deleted after the message has
558 0695 2 been delivered, unless we just finished writing an external message file.
559 0696 2 In that case, the 8-byte record is deleted when the message is deleted.
560 0697 2
561 P 0698 2 IF_ERR(PUT_SEND_MSG(8,MSGHDR[MMH_Q_DATID],.OUTRAB);,
562 0699 2 RETURN .STATUS);
563 0700 2 SAVERFA[0] = .OUTRAB[RAB$R_RFA0]; !Save the header rfa
564 0701 2 SAVERFA[1] = .OUTRAB[RAB$W_RFA4];
565 0702 2
566 0703 2 Create the message header record
567 0704 2
568 0705 2 CH$COPY(.FOLDER_DESC[DSC$W_LENGTH],.FOLDER_DESC[DSC$A_POINTER],0,
569 0706 2 MFR_S_FILEKEY,MMH[MFR_T_FILEKEY]);
570 0707 2 MMH[MFR_B_FILELEN] = .FOLDER_DESC[DSC$W_LENGTH];
571 0708 2 MSGHDR[MMH_W_FLAGS] = 0;
572 0709 2 IF .MSGFLAGS[MCR_V_SND_NEWMSG]
573 0710 2 THEN MSGHDR[MMH_V_NEWMSG] = 1;
574 0711 2 MSGHDR[MMH_V_EXTMSG] = .EXTMSG;
575 0712 2 MSGHDR[MMH_B_FLAGSIZ] = 0;
576 0713 2 MSGHDR[MMH_L_FLAGVAL] = 0;
577 0714 2 MSGHDR[MMH_B_HDSPARE] = 0;
578 0715 2 OUTRAB[RAB$L_ROP] = 0;
579 0716 2
580 0717 2 If we are to signal a 'new folder', then look for it now
581 0718 2
582 0719 2 IF .MSGFLAGS[MCR_V_TELL_NEW]
583 0720 2 THEN BEGIN
584 0721 2 OUTRAB[RAB$L_ROP] = RAB$M_NLK;
585 0722 2 OUTRAB[RAB$L_KBF] = MMH[MFR_T_FILEKEY];
586 0723 2 OUTRAB[RAB$B_KSZ] = MFR_S_FILEKEY;
587 0724 2 OUTRAB[RAB$B_KRF] = 1;
588 0725 2 IF NOT $FIND(RAB=.OUTRAB)
589 0726 2 THEN SIGNAL(MAIL$NEWFOLDER,1,.FOLDER_DESC);
590 0727 2 OUTRAB[RAB$L_ROP] = 0;
591 0728 2 OUTRAB[RAB$B_KSZ] = 8;
592 0729 2 OUTRAB[RAB$B_KRF] = 0;
593 0730 2 OUTRAB[RAB$L_KBF] = 0;
594 0731 2 END;
595 0732 2 PTR = MSGHDR[MMH_T_MISCDAT];
596 0733 2 PTR[0] = MMH_C_FROM;
597 0734 2 PTR[1] = .FROM_DESC[DSC$W_LENGTH];
598 0735 2 PTR = CH$MOVE(.PTR[1],.FROM_DESC[DSC$A_POINTER],PTR[2]);
599 0736 2 PTR[0] = MMH_C_TO;
600 0737 2 PTR[1] = .TO_DESC[DSC$W_LENGTH];
601 0738 2 PTR = CH$MOVE(.PTR[1],.TO_DESC[DSC$A_POINTER],PTR[2]);
602 0739 2 PTR[0] = MMH_C_SUBJ;
603 0740 2 PTR[1] = .SUBJ_DESC[DSC$W_LENGTH];
604 0741 2 PTR = CH$MOVE(.PTR[1],.SUBJ_DESC[DSC$A_POINTER],PTR[2]);
605 0742 2 PTR[0] = MMH_C_NREC;
606 0743 2 PTR[1] = 4;
607 0744 2 PTR1 = PTR[2];
608 0745 2 IF .EXTMSG
609 0746 2 THEN PTR1[0] = .RECOUNT !If external message, count known
610 0747 2 ELSE PTR1[0] = 0; ! otherwise store a zero until $updated
```

```

: 611      0748 2 SIZEOFF = PTR[2]-MMH;
: 612      0749 2 PTR = PTR[4];
: 613      0750 2 MMHLEN = .PTR-MMH;
: 614      0751 2
: 615      0752 2 ! Write the message header record
: 616      0753 2
: 617      P 0754 2 IF_ERR(PUT_NEW_REC(.MMHLEN,MMH,.OUTRAB);,
: 618      0755 2     RETURN .STATUS);
: 619      0756 2
: 620      0757 2 ! If message is small enough to fit into the isam file, then read the
: 621      0758 2 ! message and pack the message and write it.
: 622      0759 2
: 623      0760 2 IF NOT .EXTMSG
: 624      0761 3 THEN BEGIN
: 625      0762 3     IF .GET_RAB[RAB$V_BIO] !Ensure not connected for block mode
: 626      0763 4     THEN BEGIN
: 627      0764 4         $DISCONNECT(RAB=GET_RAB);
: 628      0765 4         GET_RAB[RAB$V_BIO] = FALSE;
: 629      0766 4         GET_RAB[RAB$W_USZ] = MAIL$K_INBUFFSZ;
: 630      0767 4         GET_RAB[RAB$L_UBF] = CNCTDESC[CNCT_T_BUFFER];
: 631      P 0768 4         IF_ERR($CONNECT(RAB=GET_RAB,ERR=UTIL$REPORT_IO_ERROR);,
: 632      0769 4             RETURN .STATUS);
: 633      0770 3     END;
: 634      0771 3     MMHRFA[0] = .OUTRAB[RAB$L_RFA0];
: 635      0772 3     MMHRFA[1] = .OUTRAB[RAB$W_RFA4];
: 636      0773 3     RECOUNT = 0;
: 637      0774 3     CH$MOVE(.MMHLEN,MMH,SAVEMMH);
: 638      0775 3
: 639      0776 3 ! Read the message, pack the records and write them to the message file
: 640      0777 3
: 641      0778 3 CH$MOVE(8,MSGHDR[MMH_Q_DATID],MMH[MFR_Q_DATIM]);
: 642      0779 3 CH$FILL(0,MFR_S_FILEKEY+1,MMH[MFR_B_FILREYLEN]);
: 643      0780 3 INIT_PACK_REC(RECLEN); !Initialize to pack records
: 644      0781 3 WHILE 1
: 645      0782 4 DO BEGIN
: 646      P 0783 4     IF_ERR((.GET_ROUTINE)(RECDESC,CNCTDESC[CNCT_T_RAB]);,
: 647      P 0784 4         IF .STATUS EQL RMS$_EOF
: 648      P 0785 4         THEN EXITLOOP
: 649      0786 4         ELSE RETURN UTIL$REPORT_IO_ERROR(CNCTDESC[CNCT_T_RAB]);
: 650      0787 4
: 651      0788 4 ! If record is terminated with a CR/LF, strip it. This type of
: 652      0789 4 ! record is produced by DEC Standard Runoff.
: 653      0790 4
: 654      0791 4 IF .RECDESC[0] GEQU 2
: 655      0792 4     AND .(.RECDESC[1] + .RECDESC[0] - 2) < 0,16>
: 656      0793 5         EQLU (CR OR LF^8)
: 657      0794 4     THEN RECDESC[0] = .RECDESC[0] - 2;
: 658      0795 4     RECOUNT = .RECOUNT + 1;
: 659      P 0796 4     IF_ERR(PACK_NEW_REC(.RECDESC[0],.RECDESC[1],
: 660      P 0797 4         RECLEN,MMH,.OUTRAB);,
: 661      0798 4         RETURN .STATUS);
: 662      0799 3     END;
: 663      0800 3
: 664      P 0801 3 IF_ERR(END_PACK(.RECLEN,MMH,.OUTRAB);,
: 665      0802 3     RETURN .STATUS);
: 666      0803 3
: 667      0804 3 ! Update the mail message header record to have the correct
```

```

668 0805 3      | record count in it
669 0806 3
670 0807 3      | OUTRAB[RAB$L_RFA0] = .MMHRFA[0];
671 0808 3      | OUTRAB[RAB$W_RFA4] = .MMHRFA[1];
672 0809 3      | OUTRAB[RAB$B_RAC] = RAB$C_RFA;
673 0810 4      | IF $FIND(RAB=.OUTRAB,ERR=UTIL$REPORT_IO_ERROR)
674 0811 4      | THEN BEGIN
675 0812 4      |     OUTRAB[RAB$W_RSZ] = .MMHLEN;
676 0813 4      |     SIZEOFF = SAVEMMH + .SIZEOFF;
677 0814 4      |     SIZECFF[0] = .RECOUNT;
678 0815 4      |     OUTRAB[RAB$L_RBF] = SAVEMMH;
679 0816 4      |     $UPDATE(RAB=.OUTRAB,ERR=UTIL$REPORT_IO_ERROR);
680 0817 4      |     END;
681 0818 3
682 0819 3      | Delete the marker record
683 0820 3
684 0821 3      | OUTRAB[RAB$L_RFA0] = .SAVERFA[0];
685 0822 3      | OUTRAB[RAB$W_RFA4] = .SAVERFA[1];
686 P 0823 3      | IF _ERR($FIND(RAB=.OUTRAB));,
687 0824 3      |     RETURN TRUE);
688 0825 3
689 0826 3      | $DELETE(RAB=.OUTRAB);
690 0827 3      | END;
691 0828 2
692 0829 2      | RETURN TRUE
693 0830 1      | END;

```

```

.EXTRN SYSS$GETTIM, SYSS$DISCONNECT
.EXTRN SYSS$CONNECT, SYSS$CREATE
.EXTRN SYSS$CLOSE, SYSS$FIND
.EXTRN SYSS$UPDATE, SYSS$DELETE

```

OFFC 00000

		5E	ED88	CE	9E	00002	
50	08	AC	0000286	8F	C1	00007	
		58		60	9E	00010	
50	08	AC		08	C1	00013	
				50	DD	00018	
50	08	AC		10	C1	0001A	
				50	DD	0001F	
50	08	AC		18	C1	00021	
				50	DD	00026	
		56	14	AC	D0	00028	
		24	A6	18	AE	9E	0002C
		20	A6	0800	8F	B0	00031
		1E	A6		01	90	00037
				04	A6	D4	0003B
		18	A6	007E10D0	8F	D0	0003E
	08	10	AC		04	E1	00046
18	AE	18	BC		09	28	0004B
				2A	11	00051	
				18	AE	9F	00053 1\$:
		00000000G	00		01	FB	00056
				18	AE	9F	0005D

```

.ENTRY MAIL$WRITE_MSG, Save R2,R3,R4,R5,R6,R7,R8,- : 0463
R9,R10,R11
MOVAB -4728(SP), SP
ADDL3 #646, CNCTDESC, R0 : 0481
MOVAB (R0), R8
ADDL3 #8, CNCTDESC, R0 : 0482
PUSHL R0
ADDL3 #16, CNCTDESC, R0 : 0483
PUSHL R0
ADDL3 #24, CNCTDESC, R0 : 0484
PUSHL R0
MOVL OUTRAB, R6 : 0510
MOVAB MMH, 36(R6)
MOVW #2048, 32(R6) : 0511
MOVB #1, 30(R6) : 0512
CLRL 4(R6) : 0513
MOVL #8261840, 24(R6) : 0514
BBC #4, MSGFLAGS, 1$ : 0518
MOVCS #9, @DATETIME, MMH : 0519
BRB 2$
PUSHAB MMH : 0521
CALLS #1, SYSS$GETTIM
PUSHAB MMH : 0522

```

				56	DD	00060	PUSHL	R6		
				02	FB	00062	CALLS	#2, MAIL\$CHECK_DATE		
				50	E8	00069	BLBS	R0, 2\$		
				56	DD	0006C	PUSHL	R6		0524
				01	FB	0006E	CALLS	#1, UTIL\$GETFILENAME		
				50	DD	00075	PUSHL	R0		
			1C	AE	9F	00077	PUSHAB	MMH		
			03	31	0007A	BRW	23\$			
			48	AE	9E	0007D	MOVAB	MMH+48, MSGHDR		0528
		5A	08	AA	9E	00081	MOVAB	8(MSGHDR), 12(SP)		0529
		OC	0C	AE	D0	00086	MOVL	12(SP), PTR		
		59	1A	AE	D0	0008A	MOVL	MMH+2, (PTR)		0530
		04	1E	AE	B0	0008E	MOVW	MMH+6, 4(PTR)		0531
		06	04	AE	B0	00093	MOVW	#4, 6(PTR)		0532
			10	AE	D4	00097	CLRL	EXTMSG		0537
				3C	C1	0009A	ADDL3	#60, CNCTDESC, R0		0538
				60	D1	0009F	CMP	(R0), MAIL\$L_SIZLIM		
				03	1A	000A6	BGTRU	3\$		
				02	C5	31	BRW	21\$		
				01	D0	000AB	MOVL	#1, EXTMSG		0540
				A6	D0	000AF	MOVL	60(R6), OUTNAM		0541
				24	A7	000B3	MOVL	36(OUTNAM), PROXAB		0542
				1F	12	000B7	BNEQ	4\$		0544
				00	2C	000B9	MOVCS	#0, (SP), #0, #88, \$RMS_PTR		0547
				FE94	CD	000C0				
				5813	8F	B0	MOVW	#22547, \$RMS_PTR		
				00000000G	00	D0	MOVL	MAIL\$L_MYUIC, TMPXABPRO+12		0548
				FE94	CD	9E	MOVAB	TMPXABPRO, PROXAB		0549
				04	AB	34	MNEGW	#52, 8(PROXAB)		0551
				04	AB	D4	CLRL	4(PROXAB)		0552
				28	A7	D0	MOVL	40(OUTNAM), OUTNAM		0553
				39	A7	9A	MOVZBL	57(OUTNAM), R0		0554
				3A	A7	9A	MOVZBL	58(OUTNAM), R1		
				51	C1	000EB	ADDL3	R1, R0, DEVDIRDESC		
				50	A7	D0	MOVL	68(OUTNAM), DEVDIRDESC+4		0555
				F4	AD	44	MOVZBL	#255, RECDISC		0556
				E0	AD	FF	MOVAB	NAMBUF, RECDISC+4		0557
				E4	AD	FD94	PUSHAB	DEVDIRDESC		0558
				FO	AD	9F	PUSHL	16(SP)		
				10	AE	DD	PUSHL	R6		
				56	DD	00106	CALLS	#3, MAIL\$CHECK_DATE		
				03	FB	00108	BLBS	R0, 5\$		
				50	E8	0010F	BRW	22\$		
				02	6A	31	PUSHL	12(SP)		0566
				OC	BE	DD	PUSHL	12(MSGHDR)		0565
				OC	AA	DD	PUSHAB	DEVDIRDESC		0564
				FO	AD	9F	PUSHAB	RECDISC		
				EO	AD	9F	PUSHAB	RECDISC		
				EO	AD	9F	PUSHAB	RECDISC		
				00000000G	JO	9F	PUSHAB	MAIL\$SD EXTMSGNAM		
				00000000G	06	FB	CALLS	#6, SYS\$FAO		
				00	2C	00131	MOVCS	#0, (SP), #0, #80, \$RMS_PTR		0577
				90	AD	00138				
				5003	8F	B0	MOVW	#20483, \$RMS_PTR		
				40	8F	9A	MOVZBL	#64, \$RMS_PTR+4		
				50	3C	C1	ADDL3	#60, CNCTDESC, R0		
				08	AD	60	MOVL	(R0), \$RMS_PTR+16		
				A0	AD	11	MOVW	#17, \$RMS_PTR+22		
				A6	AD	90	MOVW	#17, \$RMS_PTR+22		

		A8	AD	007E10A2	8F	D0	00152	MOVL	#8261794, \$RMS_PTR+24		
		AE	AD	0202	8F	B0	0015A	MOVW	#514, \$RMS_PTR+30		
		B4	AD		5B	D0	00160	MOVL	PROXAB, \$RMS_PTR+36		
		B8	AD	FEEC	CD	9E	00164	MOVAB	TMPNAM, \$RMS_PTR+40		
		BC	AD	E4	AD	D0	0016A	MOVL	RECDESC+4, \$RMS_PTR+44		
		C4	AD	E0	AD	90	0016F	MOVW	RECDESC, \$RMS_PTR+52		
		11	AC		01	E1	00174	BBC	#1, MSGFLAGS+T, 6\$	0579	
DA	AD	06	00		01	F0	00179	INSV	#1, #0, #2, TMPFAB+74	0580	
		02	50	3C	AB	D0	0017F	6\$: MOVL	60(R8), TFAB	0585	
		4D	10	AC	06	E1	00183	BBC	#6, MSGFLAGS, 8\$	0590	
			02	1F	A0	91	00188	CMPB	31(TFAB), #2	0591	
					06	13	0018C	BEQL	7\$		
			03	1F	A0	91	0018E	CMPB	31(TFAB), #3		
					41	12	00192	BNEQ	8\$		
		3C	16	A0	06	E1	00194	7\$: BBC	#6, 22(TFAB), 8\$	0598	
			AE	AD	A0	B0	00199	MOVW	30(TFAB), TMPFAB+30	0600	
		32	05	AB	03	E0	0019E	BBS	#3, 5(R8), 8\$	0602	
					58	DD	001A3	PUSHL	R8	0604	
			00000000G	00	01	FB	001A5	CALLS	#1, SYSSDISCONNECT		
			05	A8	08	88	001AC	BISB2	#8, 5(R8)	0605	
			24	A8	00	D0	001B0	MOVL	MAIL\$GL BIGBUFFER, 36(R8)	0606	
20	AB	00000000G	00	0200	8F	A5	001B8	MULW3	#512, MAIL\$GL BIGBUFSIZ, 32(R8)	0607	
				00000000G	00	9F	001C3	PUSHAB	UTIL\$REPORT_IO_ERROR	0609	
					58	DD	001C9	PUSHL	R8		
			00000000G	00	02	FB	001CB	CALLS	#2, SYSSCONNECT		
				6B	50	E9	001D2	BLBC	STATUS, 9\$		
0044	8F	00	6E		00	2C	001D5	8\$: MOVCS	#0, (SP), #0, #68, \$RMS_PTR	0615	
				FF4C	CD	001DC					
				4401	8F	B0	001DF	MOVW	#17409, \$RMS_PTR		
			FF4C	CD	8F	D0	001E6	MOVL	#8261794, \$RMS_PTR+24		
			FF64	CD	8F	D0	001E6	MOVL	#8261794, \$RMS_PTR+24		
			88	AD	AD	9E	001EF	MOVAB	TMPFAB, \$RMS_PTR+60		
				01	03	EF	001F4	EXTZV	#3, #1, 5(R8), R0	0617	
				01	50	F0	001FA	INSV	R0, #3, #1, TMPRAB+5		
FF51	50	05	A8	01	50	F0	00201	INSV	R0, #6, #1, TMPFAB+22		
	CD		01	03	50	F0	00201	INSV	R0, #6, #1, TMPFAB+22		
A6	AD		01	06	00	2C	00207	MOVCS	#0, (SP), #0, #96, \$RMS_PTR	0623	
0060	8F		00	6E	00	2C	00207	MOVCS	#0, (SP), #0, #96, \$RMS_PTR	0623	
					CD	0020E					
				FEEC	8F	B0	00211	MOVW	#24578, \$RMS_PTR		
				6002	01	8E	00218	MNEGB	#1, \$RMS_PTR+2		
					CD	9E	0021D	MOVAB	NAMBUF, \$RMS_PTR+4		
					01	8E	00224	MNEGB	#1, \$RMS_PTR+10		
					CD	9E	00229	MOVAB	NAMBUF, \$RMS_PTR+12		
					00	9F	00230	PUSHAB	UTIL\$REPORT_IO_ERROR	0626	
					AD	9F	00236	PUSHAB	TMPFAB		
			00000000G	00	02	FB	00239	CALLS	#2, SYSSCREATE		
				01	50	E8	00240	9\$: BLBS	STATUS, 10\$		
					04	00243		RET			
					03	E1	00244	10\$: BBC	#3, TMPFAB+64, 11\$	0627	
04	D0	AD	20	D3	AD	E9	00249	BLBC	TMPFAB+67, 12\$	0628	
				90	AD	9F	0024D	11\$: PUSHAB	TMPFAB	0630	
					01	FB	00250	CALLS	#1, SYSSCLOSE		
			00000000G	00	8F	3C	00257	MOVZWL	#324, TMPFAB+8	0631	
			98	AD	AD	D4	0025D	CLRL	TMPFAB+12	0632	
					8F	3C	00260	MOVZWL	#324, -(SP)	0633	
				7E	8F	3C	00260	MOVZWL	#324, -(SP)	0633	
			00000000G	00	01	FB	00265	CALLS	#1, LIB\$SIGNAL		
					04	0026C		RET			
					00	9F	0026D	12\$: PUSHAB	UTIL\$REPORT_IO_ERROR	0639	
					CD	9F	00273	PUSHAB	TMPRAB		
					FF4C	CD	9F	00273	PUSHAB	TMPRAB	

00000000G	00	02	FB	00277	CALLS	#2, SYSSCONNECT	
	52	50	DO	0027E	MOVL	R0, STATUS	
	5F	52	E9	00281	BLBC	STATUS, 16\$	
FF64	CD	5B	D4	00284	CLRL	RECOUNT	0640
		8F	DO	00286	MOVL	#8261840, TMPRAB+24	0641
		58	DD	0028F	PUSHL	R8	0652
		AD	9F	00291	PUSHAB	RECDESC	
04	BC	02	FB	00294	CALLS	#2, @GET ROUTINE	
		50	DO	00298	MOVL	R0, STATUS	
		52	E8	0029B	BLBS	STATUS, 14\$	
0001827A	8F	52	D1	0029E	CMPL	STATUS, #98938	
		53	13	002A5	BEQL	18\$	
		58	DD	002A7	PUSHL	R8	
00000000G	00	01	FB	002A9	CALLS	#1, UTIL\$REPORT_IO_ERROR	
18	FF51	31	11	002B0	BRB	16\$	
		03	E0	002B2	BBS	#3, TMPRAB+5, 15\$	0653
		AD	D1	002B8	CMPL	RECDESC, #2	0654
		12	1F	002BC	BLSSU	15\$	
50	E4	AD	C1	002BE	ADDL3	RECDESC, RECDESC+4, R0	0655
	OA0D	8F	B1	002C4	CMPL	-2(R0), #2573	0656
		04	12	002CA	BNEQ	15\$	
	E0	AD	C2	002CC	SUBL2	#2, RECDESC	0657
		CD	9F	002D0	PUSHAB	TMPRAB	0662
		AD	7D	002D4	MOVQ	RECDESC, -(SP)	
	FC44	CF	FB	002D8	CALLS	#3, PUT_SEND_MSG	
		52	DO	002DD	MOVL	R0, STATUS	
		52	E8	002E0	BLBS	STATUS, 17\$	
		8F	88	002E3	BISB2	#128, TMPFAB+5	
	95	AD	9F	002E8	PUSHAB	TMPFAB	
					CALLS	#1, SYSSCLOSE	
00000000G	00	01	FB	002EB	CALLS	#1, SYSSCLOSE	
	50	52	DO	002F2	MOVL	STATUS, R0	
		04	06	002F5	RET		
		5B	D6	002F6	INCL	RECOUNT	0663
		95	11	002F8	BRB	13\$	0642
56	FF51	03	E1	002FA	BBC	#3, TMPRAB+5, 20\$	0669
50	08	8F	C1	00300	ADDL3	#100, CNCTDESC, R0	0670
		60	DO	00309	MOVL	(R0), RECOUNT	
		48	12	0030C	BNEQ	20\$	
		CD	9F	0030E	PUSHAB	TMPRAB	0672
		01	FB	00312	CALLS	#1, SYSSDISCONNECT	
		08	8A	00319	BICB2	#8, TMPRAB+5	0673
		8F	B0	0031E	MOVW	#512, TMPRAB+32	0674
FF70	CD	8F	C1	00325	ADDL3	#134, CNCTDESC, TMPRAB+36	0675
		00	9F	00330	PUSHAB	UTIL\$REPORT_IO_ERROR	0676
		CD	9F	00336	PUSHAB	TMPRAB	
		02	FB	0033A	CALLS	#2, SYSSCONNECT	
		50	E9	00341	BLBC	R0, 20\$	
		CD	9F	00344	PUSHAB	TMPRAB	0677
		01	FB	00348	CALLS	#1, SYSSGET	
		50	E9	0034F	BLBC	R0, 20\$	
		5B	D6	00352	INCL	RECOUNT	0678
		EE	11	00354	BRB	19\$	
		8F	DO	00356	MOVL	#8261720, TMPFAB+24	0680
	A8	AD	9F	0035E	PUSHAB	UTIL\$REPORT_IO_ERROR	0681
		AD	9F	00364	PUSHAB	TMPFAB	
		02	FB	00367	CALLS	#2, SYSSCLOSE	
00000000G	00	34	11	0036E	BRB	24\$	0538

				0C	AE DD 00370	21\$:	PUSHL	12(SP)	0686
					56 DD 00373		PUSHL	R6	
		00000000G	00		02 FB 00375		CALLS	#2, MAIL\$CHECK_DATE	
			25		50 E8 0037C		BLBS	R0, 24\$	
		00000000G	00		56 DD 0037F	22\$:	PUSHL	R6	0689
					01 FB 00381		CALLS	#1, UTIL\$GETFILENAME	
				10	50 DD 00388		PUSHL	R0	
					AE DD 0038A		PUSHL	16(SP)	0688
					02 DD 0038D	23\$:	PUSHL	#2	
		00000000G	00	007E80A2	8F DD 0038F		PUSHL	#8290466	
			50	007E80A2	04 FB 00395		CALLS	#4, LIB\$SIGNAL	
					8F DD 0039C		MOVL	#8290466, R0	0690
					04 003A3		RET		
					56 DD 003A4	24\$:	PUSHL	R6	0699
				10	AE DD 003A6		PUSHL	16(SP)	
					08 DD 003A9		PUSHL	#8	
		FB71	CF		03 FB 003AB		CALLS	#3, PUT_SEND_MSG	
			01		50 E8 003B0		BLBS	STATUS, 25\$	
					04 003B3		RET		
		FB	AD	10	A6 DD 003B4	25\$:	MOVL	16(R6), SAVERFA	0700
		FC	AD	14	A6 3C 003B9		MOVZWL	20(R6), SAVERFA+4	0701
			57	0C	AC DD 003BE		MOVL	FOLDER_DESC, R7	0705
27	00	04	B7		67 2C 003C2		MOVCS	(R7), #4(R7), #0, #39, MMH+9	0706
				21	AE 003C8				
		20	AE		67 90 003CA		MOVB	(R7), MMH+8	0707
					6A B4 003CE		CLRW	(MSGHDR)	0708
				10	AC E9 003D0		BLBC	MSGFLAGS, 26\$	0709
					01 88 003D4		BISB2	#1, (MSGHDR)	0710
6A	01		03		AE FO 003D7	26\$:	INSV	EXTMSG, #3, #1, (MSGHDR)	0711
					02 AA 94 003DD		CLRB	2(MSGHDR)	0712
					03 AA D4 003E0		CLRL	3(MSGHDR)	0713
					07 AA 94 003E3		CLRB	7(MSGHDR)	0714
					04 A6 D4 003E6		CLRL	4(R6)	0715
		3A	10	AC	05 E1 003E9		BBC	#5, MSGFLAGS, 28\$	0719
			04	A6	8F DD 003EE		MOVL	#1048576, 4(R6)	0721
			30	A6	AE 9E 003F6		MOVAB	MMH+9, 48(R6)	0722
			34	A6	8F B0 003FB		MOVW	#295, 52(R6)	0723
				0127	56 DD 00401		PUSHL	R6	0725
		00000000G	00		01 FB 00403		CALLS	#1, SYSS\$FIND	
			11		50 E8 0040A		BLBS	R0, 27\$	
					57 DD 0040D		PUSHL	R7	0726
					01 DD 0040F		PUSHL	#1	
					8F DD 00411		PUSHL	#8306347	
		00000000G	00	007EBEAB	03 FB 00417		CALLS	#3, LIB\$SIGNAL	
					A6 D4 0041E	27\$:	CLRL	4(R6)	0727
		34	A6		08 B0 00421		MOVW	#8, 52(R6)	0728
					A6 D4 00425		CLRL	48(R6)	0730
			59	10	AA 9E 00428	28\$:	MOVAB	16(R10), PTR	0732
					69 B4 0042C		CLRW	(PTR)	0733
			02	A9	BE B0 0042E		MOVW	@8(SP), 2(PTR)	0734
		04	57	08	AE C1 00433		ADDL3	#4, 8(SP), R7	0735
			A9		A9 28 00438		MOVCS	2(PTR), @ (R7)+, 4(PTR)	
					53 DD 0043E		MOVL	R3, PTR	
					01 B0 00441		MOVW	#1, (PTR)	0736
			02	A9	BE B0 00444		MOVW	@4(SP), 2(PTR)	0737
		04	57	04	AE C1 00449		ADDL3	#4, 4(SP), R7	0738
04	A9		97	02	A9 28 0044E		MOVCS	2(PTR), @ (R7)+, 4(PTR)	

			59		53	D0	00454		MOVL	R3, PTR		
			69		02	B0	00457		MOVW	#2, (PTR)		0739
		02	A9	00	BE	B0	0045A		MOVW	@0(SP), 2(PTR)		0740
	04	57	6E		04	C1	0045F		ADDL3	#4, (SP), R7		0741
		A9	97	02	A9	28	00463		MOVW	2(PTR), @ (R7)+, 4(PTR)		
			59		53	D0	00469		MOVL	R3, PTR		
			69	00040005	8F	D0	0046C		MOVL	#262149, (PTR)		0742
			53	04	A9	9E	00473		MOVAB	4(R9), R3		0744
			50		53	D0	00477		MOVL	R3, PTR1		
			05	10	AE	E9	0047A		BLBC	EXTMSG, 29\$		0746
			60		5B	D0	0047E		MOVL	RECOUNT, (PTR1)		
					02	11	00481		BRB	30\$		
					60	D4	00483	29\$:	CLRL	(PTR1)		0747
			50	18	AE	9E	00485	30\$:	MOVAB	MMH, R0		0748
		SA	53		50	C3	00489		SUBL3	R0, R3, SIZEOFF		
			59		08	C0	0048D		ADDL2	#8, PTR		0749
			50	18	AE	9E	00490		MOVAB	MMH, R0		0750
			59		50	C2	00494		SUBL2	R0, MMHLEN		
					56	DD	00497		PUSHL	R6		0755
				1C	AE	9F	00499		PUSHAB	MMH		
					59	DD	0049C		PUSHL	MMHLEN		
		FAB5	CF		03	FB	0049E		CALLS	#3, PUT_NEW_REC		
			38		50	E9	004A3		BLBC	STATUS, 32\$		
			03	10	AE	E9	004A6		BLBC	EXTMSG, 31\$		0760
					011C	31	004AA		BRW	39\$		
					03	E1	004AD	31\$:	BBC	#3, 5(R8), 33\$		0762
					58	DD	004B2		PUSHL	R8		0764
			00		01	FB	004B4		CALLS	#1, SYSSDISCONNECT		
			05		08	8A	004BB		BICB2	#8, 5(R8)		0765
			20		8F	B0	004BF		MOVW	#512, 32(R8)		0766
			08	0200	8F	C1	004C5		ADDL3	#134, CNCTDESC, 36(R8)		0767
				00000086	00	9F	004CF		PUSHAB	UTIL\$REPORT_IO_ERROR		0769
				00000000G	58	DD	004D5		PUSHL	R8		
			00		02	FB	004D7		CALLS	#2, SYSSCONNECT		
			01		50	E8	004DE	32\$:	BLBS	STATUS, 33\$		
					04	004E1			RET			
			E8	AD	A6	D0	004E2	33\$:	MOVL	16(R6), MMHRFA		0771
			EC	AD	A6	3C	004E7		MOVZWL	20(R6), MMHRFA+4		0772
					5B	D4	004EC		CLRL	RECOUNT		0773
					59	28	004EE		MOVW	MMHLEN, MMH, SAVEMMH		0774
					08	28	004F5		MOVW	#8, @12(SP), MMH		0778
					00	2C	004FB		MOVW	#0, (SP), #0, #40, MMH+8		0779
					20	AE	00500					
					14	AE	9F	00502	PUSHAB	RECLN		0780
					01	FB	00505	34\$:	CALLS	#1, INIT_PACK_REC		
					58	DD	0050A		PUSHL	R8		0786
					AD	9F	0050C		PUSHAB	RECDESC		
			04	BC	02	FB	0050F		CALLS	#2, @GET_ROUTINE		
			13		50	E8	00513		BLBS	STATUS, 35\$		
				0001827A	50	D1	00516		CMPL	STATUS, #98938		
					39	13	0051D		BEQL	37\$		
					58	DD	0051F		PUSHL	R8		
				00000000G	01	FB	00521		CALLS	#1, UTIL\$REPORT_IO_ERROR		
					04	00528			RET			
					AD	D1	00529	35\$:	CMPL	RECDESC, #2		0791
			02	E0	12	1F	0052D		BLSSU	36\$		
					AD	C1	0052F		ADDL3	RECDESC, RECDESC+4, R0		0792

0A0D	8F	FE	A0	B1	00535		CMPW	-2(R0), #2573	0793
			04	12	0053B		BNEQ	36\$	
E0	AD		02	C2	0053D		SUBL2	#2, RECDISC	0794
			5B	D6	00541	36\$:	INCL	RECOUNT	0795
			56	DD	00543		PUSHL	R6	0798
		1C	AE	9F	00545		PUSHAB	MMH	
		1C	AE	9F	00548		PUSHAB	RECLN	
FA33	7E	EO	AD	7D	0054B		MOVQ	RECDISC, -(SP)	
	CF		05	FB	0054F		CALLS	#5, PACK_NEW_REC	
	B3		50	E8	00554		BLBS	STATUS, 34\$	
				04	00557		RET		
			56	DD	00558	37\$:	PUSHL	R6	0802
		1C	AE	9F	0055A		PUSHAB	MMH	
		1C	AE	DD	0055D		PUSHL	RECLN	
FA07	CF		03	FB	00560		CALLS	#3, END_PACK	
	64		50	E9	00565		BLBC	STATUS, -40\$	
10	A6	E8	AD	D0	00568		MOVL	MMHRFA, 16(R6)	0807
14	A6	EC	AD	B0	0056D		MOVW	MMHRFA+4, 20(R6)	0808
1E	A6		02	90	00572		MOVB	#2, 30(R6)	0809
		00000000G	00	9F	00576		PUSHAB	UTIL\$REPORT_IO_ERROR	0810
			56	DD	0057C		PUSHL	R6	
00000000G	00		02	FB	0057E		CALLS	#2, SYSS\$FIND	
	22		50	E9	00585		BLBC	R0, 38\$	
22	A6		59	B0	00588		MOVW	MMHLEN, 34(R6)	0812
	5A	0818	CE4A	9E	0058C		MOVAB	SAVEMMH[SIZEOFF], SIZEOFF	0813
	6A		5B	D0	00592		MOVL	RECOUNT, (SIZEOFF)	0814
28	A6	0818	CE	9E	00595		MOVAB	SAVEMMH, 40(R6)	0815
		00000000G	00	9F	0059B		PUSHAB	UTIL\$REPORT_IO_ERROR	0816
			56	DD	005A1		PUSHL	R6	
00000000G	00		02	FB	005A3		CALLS	#2, SYSS\$UPDATE	
10	A6	F8	AD	D0	005AA	38\$:	MOVL	SAVERFA, 16(R6)	0821
14	A6	FC	AD	B0	005AF		MOVW	SAVERFA+4, 20(R6)	0822
			56	DD	005B4		PUSHL	R6	0824
00000000G	00		01	FB	005B6		CALLS	#1, SYSS\$FIND	
	09		50	E9	005BD		BLBC	STATUS, 39\$	
			56	DD	005C0		PUSHL	R6	0826
00000000G	00		01	FB	005C2		CALLS	#1, SYSS\$DELETE	
	50		01	D0	005C9	39\$:	MOVL	#1, R0	0829
			04	005CC	40\$:		RET		0830

; Routine Size: 1485 bytes, Routine Base: \$CODE\$ + 00DF

```

: 695      0831  1 ROUTINE SEND_MSG (CNCTDESC,DEVDIR_DESC,UIC,SND_FAB) =
: 696      0832  1 :++
: 697      0833  1 :-- This routine actually sends the message
: 698      0834  1 :--
: 699      0835  2 BEGIN
: 700      0836  2 MAP
: 701      0837  2     CNCTDESC : REF $BLOCK,
: 702      0838  2     DEVDIR_DESC : REF $BLOCK,
: 703      0839  2     SND_FAB : REF $BLOCK;
: 704      0840  2
: 705      0841  2 BIND
: 706      0842  2     GET_RAB = CNCTDESC[CNCT_T_RAB] : $BLOCK;
: 707      0843  2
: 708      0844  2 LOCAL
: 709      0845  2     FF_FLAG,                               ! Form-feed flag
: 710      0846  2     STATUS,                               ! Routine status
: 711      0847  2     SAVE_SFS,
: 712      0848  2     SAVE_STV,
: 713      0849  2     SND_RAB : $RAB_DECL,
: 714      0850  2     SND_NAM : $NAM_DECL,
: 715      0851  2     SND_XABPRO : $XABPRO_DECL,
: 716      0852  2     SND_ESNAM_BUF : $BLOCK[NAM$C_MAXRSS],
: 717      0853  2     DEF_BUFF: VECTOR[NAM$C_MAXRSS, BYTE],       ! Default name buffer
: 718      0854  2     REC_BUFF: VECTOR[MAIL$R_INBUFSZ, BYTE],  ! Record buffer
: 719      0855  2     REC_SIZ,                               ! Record size
: 720      0856  2     DESC: $BLOCK[8];                       ! String descriptor
: 721      0857  2
: 722      0858  2 INIT_SDESC(DESC);
: 723      0859  2
: 724      0860  2 : Create and set the file name and file owner
: 725      0861  2
: 726      0862  2 CH$COPY(.DEVDIR_DESC[DSC$W_LENGTH],           !Concatenate dev,dir
: 727      0863  2     .DEVDIR_DESC[DSC$A_POINTER],
: 728      0864  2     %CHARCOINT(DEF$FILENAME,DEF$FILETYPE),    ! and filename & type
: 729      0865  2     UPLIT_BYTE(DEF$FILENAME,DEF$FILETYPE),
: 730      0866  2     0,NAM$C_MAXRSS,DEF_BUFF);
: 731      0867  2
: 732      0868  2 DESC[DSC$W_LENGTH] = .DEVDIR_DESC[DSC$W_LENGTH] +
: 733      0869  2     %CHARCOINT(DEF$FILENAME,DEF$FILETYPE);
: 734      0870  2 DESC[DSC$A_POINTER] = DEF_BUFF;
: 735      0871  2
: 736      0872  2 : First try to open the file as an ISAM file
: 737      0873  2
: 738      P 0874  2 $NAM_INIT(NAM=SND_NAM,
: 739      P 0875  2     ESS=NAM$C_MAXRSS,
: 740      P 0876  2     ESA=SND_ESNAM_BUF,
: 741      P 0877  2     RSS=NAM$C_MAXRSS,
: 742      0878  2     RSA=SND_ESNAM_BUF);
: 743      0879  2
: 744      0880  2 STATUS = MAIL$CREATE_ISAM(.SND_FAB,SND_RAB,SND_NAM,DESC,.UIC,
: 745      0881  2     (MCR_M_CIF OR MCR_M_EXEC));
: 746      0882  2
: 747      0883  2 IF (.STATUS
: 748      0884  2     AND (.SND_FAB[FAB$B_ORG] EQL FAB$C_IDX))
: 749      0885  2 THEN BEGIN
: 750      P 0886  2     IF_ERR(MAIL$WRITE_MSG(GET_RECORD,.CNCTDESC,MAIL$SD_NEWMAIL,
: 751      P 0887  2     (MCR_M_EXEC OR MCR_M_SND_NEWMSG OR MCR_M_CNCT_RAB),
```

```

752 P 0888 3
753 0889
754 0890
755 0891
756 0892
757 0893
758 0894
759 0895
760 0896
761 0897
762 0898
763 0899
764 0900
765 0901
766 0902
767 0903
768 0904
769 0905
770 0906
771 0907
772 0908
773 0909
774 0910
775 0911
776 0912
777 P 0913
778 0914
779 0915
780 0916
781 0917
782 0918
783 0919
784 0920
785 0921
786 0922
787 0923
788 0924
789 0925
790 0926
791 0927
792 0928
793 0929
794 0930
795 0931
796 0932
797 0933
798 0934
799 0935
800 0936
801 0937
802 0938
803 0939
804 0940
805 P 0941
806 P 0942
807 P 0943
808 0944

                                SND_RAB);,
                                RETURN .STATUS);
END
ELSE BEGIN
SAVE_STS = .SND_FAB[FAB$S_STS];
SAVE_STV = .SND_FAB[FAB$S_STV];
IF .SND_FAB[FAB$W_IFI] NEQ 0
    THEN $CLOSE(FAB=.SND_FAB);           !Close in case
SND_FAB[FAB$S_STS] = .SAVE_STS;
SND_FAB[FAB$S_STV] = .SAVE_STV;
! If locked ISAM file, signal error and return
IF ((.STATUS EQL RMSS_FLK)
    AND (.SND_FAB[FAB$B_ORG] EQL FAB$C_IDX))
THEN RETURN UTIL$REPORT_IO_ERROR(.SND_FAB);

IF .STATUS NEQ RMSS_PRV
THEN BEGIN
SND_FAB[FAB$B_SHR] = 0;
SND_FAB[FAB$B_ORG] = FAB$C_SEQ;
SND_FAB[FAB$B_FAC] = FAB$M_PUT OR FAB$M_TRN;
SND_FAB[FAB$B_RAT] = FAB$M_CR;
SND_FAB[FAB$B_RFM] = FAB$C_VAR;
SND_FAB[FAB$S_XAB] = SND_XABPRO;
$XABPRO_INIT(XAB=SND_XABPRO,
             NXT=0);
SND_XABPRO[XAB$S_UIC] = .UIC;
SND_XABPRO[XAB$W_PRO] = SND_PRO;
! Open/create the addressee's mail file
STATUS = $CREATE(FAB = .SND_FAB);
IF .STATUS EQL RMSS_FLK           ! If file locked,
THEN BEGIN                       ! unconditionally create
SND_FAB[FAB$V_CIF] = 0;
STATUS = $CREATE(FAB = .SND_FAB);
END;

IF NOT .SND_FAB[$FAB_DEV(DIR)]
OR .SND_FAB[$FAB_DEV(FOR)]
THEN BEGIN
$CLOSE(FAB=.SND_FAB);
STATUS = $$$_IVDEVNAM;
SND_FAB[FAB$S_STS] = $$$_IVDEVNAM;
SND_FAB[FAB$S_STV] = 0;
END;
IF .SND_XABPRO[XAB$S_UIC] NEQ .UIC ! Make sure file owner
THEN STATUS = RMSS_PRV;         ! is really correct
END;
IF NOT .STATUS                   ! Check OPEN/CREATE
THEN RETURN UTIL$REPORT_IO_ERROR(.SND_FAB);

$RAB_INIT(RAB=SND_RAB,
          FAB=.SND_FAB,
          CIX=MAIL$_OPENOUT,
          ROP=EOF);
```

```
0945  
0946  
0947  
0948  
0949  
0950  
0951  
0952  
0953  
0954  
0955  
0956  
0957  
0958  
0959  
0960  
0961  
0962  
0963  
P 0964  
0965  
0966  
0967  
0968  
0969  
0970  
P 0971  
P 0972  
P 0973  
P 0974  
0975  
P 0976  
0977  
0978  
P 0979  
P 0980  
P 0981  
0982  
P 0983  
0984  
0985  
0986  
P 0988  
P 0989  
P 0990  
P 0991  
P 0992  
0993  
0994  
0995  
P 0996  
0997  
0998  
0999  
1000  
1001
```

```
STATUS = $CONNECT(RAB = SND_RAB,ERR=UTIL$REPORT_IO_ERROR);  
IF NOT .STATUS  
  THEN RETURN .STATUS;  
SND_RAB[RAB$$_CTX] = MAIL$WRITEERR;  
:  
: Make sure that the input RAB is in record rather than block mode  
:  
IF .GET_RAB[RAB$$_BIO]  
  THEN BEGIN  
  $DISCONNECT(RAB=GET_RAB);  
  GET_RAB[RAB$$_BIO] = FALSE;  
  GET_RAB[RAB$$_USZ] = MAIL$K_INBUFFSZ;  
  GET_RAB[RAB$$_UBF] = CNCTDESC[CNCT_T_BUFFER];  
  $CONNECT(RAB=GET_RAB,ERR=UTIL$REPORT_IO_ERROR);  
  END;  
:  
: Create the message delimiter  
IF_ERR(PUT_SEND_MSG(1,UPLIT(%CHAR(FF)),SND_RAB);,  
  RETURN .STATUS);  
:  
: Create the message header  
DESC[DESC$$_LENGTH] = MAIL$K_INBUFFSZ;  
DESC[DESC$$_POINTER] = REC_BUFF;  
$FAO(Descriptor('From: !_!AS !17%D *'), ! Format From: line  
  REC_SIZ,  
  DESC,  
  CNCTDESC[CNCT_Q_FROMDESC],  
  0);  
IF_ERR(PUT_SEND_MSG(.REC_SIZ,REC_BUFF,SND_RAB);, ! Put FROM: line  
  RETURN .STATUS);  
$FAO(MAIL$GETMSG(MAIL$MAIL10), !Put To: line  
  REC_SIZ,  
  DESC,  
  CNCTDESC[CNCT_Q_TODESC]);  
IF_ERR(PUT_SEND_MSG(.REC_SIZ,REC_BUFF,SND_RAB);, ! Put TO: line  
  RETURN .STATUS);  
IF .(CNCTDESC[CNCT_Q_SUBJECTDESC]<0,16,0> NEQ 0 ! If subject specified  
  THEN BEGIN  
  $FAO(MAIL$GETMSG(MAIL$MAIL20), ! Format subject line  
    REC_SIZ,  
    DESC,  
    CNCTDESC[CNCT_Q_SUBJECTDESC]);  
  IF_ERR(PUT_SEND_MSG(.REC_SIZ,REC_BUFF,SND_RAB);,  
    RETURN .STATUS);  
  END;  
IF_ERR(PUT_SEND_MSG(0,REC_BUFF,SND_RAB);, ! Put a blank line  
  RETURN .STATUS);  
:  
: Copy the message text to the file  
FF_FLAG = 0;
```

```

866      1002  3  WHILE 1
867      1003  4  DO BEGIN
868      P 1004  4    IF_ERR($GET(RAB =GET_RAB,ERR=UTIL$REPORT_IO_ERROR);,
869      P 1005  4      IF .STATUS NEQ RMSS EOF
870      P 1006  4      THEN RETURN .STATUS;
871      1007  4      RETURN TRUE);
872      1008  4    IF .FF_FLAG
873      1009  5    THEN BEGIN
874      1010  5      : If this line is 'From', then insert a blank line in between
875      1011  5      :
876      1012  5      IF .GET_RAB[RAB$W_RSZ] GTRU %CHARCOUNT('From')
877      1013  5      AND .(.GET_RAB[RAB$L_RBF]) EQL 'From'
878      1014  5      THEN IF_ERR(PUT_SEND_MSG(0,.GET_RAB[RAB$L_RBF],SND_RAB);,
879      P 1015  5      RETURN .STATUS);
880      1016  5      FF_FLAG = 0; !No longer following FF
881      1017  5    END
882      1018  5  ELSE IF .GET_RAB[RAB$W_RSZ] EQL 1
883      1019  4  AND .(.GET_RAB[RAB$L_RBF]) <0,8> EQL FF
884      1020  4  THEN FF_FLAG = 1;
885      1021  4  :
886      1022  4  : If record is terminated with a CR/LF, strip it. This type of
887      1023  4  : record is produced by DEC Standard Runoff.
888      1024  4  :
889      1025  4  IF .GET_RAB[RAB$W_RSZ] GEQU 2
890      1026  4  AND .(.GET_RAB[RAB$L_RBF] + .GET_RAB[RAB$W_RSZ] - 2) <0,16> EQLU
891      1027  4  (CR OR LF^8)
892      1028  5  THEN GET_RAB[RAB$W_RSZ] = .GET_RAB[RAB$W_RSZ] - 2;
893      1029  4  :
894      1030  4  IF_ERR(PUT_SEND_MSG(.GET_RAB[RAB$W_RSZ],.GET_RAB[RAB$L_RBF],SND_RAB)..
895      P 1031  4  RETURN .STATUS);
896      1032  4  END;
897      1033  3  END;
898      1034  2  END; !Of old format mail file
899      1035  2
900      1036  2  RETURN TRUE
901      1037  1  END;
  
```

```

31 21 20 20 20 53 41 21 5F 21 3A 6D 6F 72 46 006AC P.AAA: .ASCII \MAIL\
2A 20 44 25 37 006B0 .ASCII \.MAIL\
00 00 00 0C 006B4 P.AAB: .ASCII <12><0><0><0>
6D 6F 72 46 006B8 P.AAD: .ASCII \From: !_!AS !17%D *\
2A 20 44 25 37 006C7
00000014 006CC P.AAC: .LONG 20
00000000' 006D0 .ADDRESS P.AAD
  
```

OFFC 0000 SEND_MSG:

```

5E FAF8 CE 9E 00002 .WORD Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11 : 0831
58 04 AC D0 00007 MOVAB -1288(SP), SP :
57 0286 C8 9E 0000B MOVL CNCTDESC, R8 : 0842
04 AE 7C 00010 MOVAB 646(R8), R7 :
56 08 AC D0 00013 CLRQ DESC : 0858
5B 66 3C 00017 MOVL DEVDIR_DESC, R6 : 0862
MOVZWL (R6), R11 :
  
```

SA	00	04	5A 59 B6	FF 020C	8F 9A 9E 5B 2C 69 0D 18 0002A	0001A 0001E 00023 00029 0002C 0002F 00032 00038	MOVZBL #255, R10 MOVAB DEF_BUFF, R9 MOVCS R11, 24(R6), #0, R10, (R9)		
SA	00	A2	5A 5A AF		5B 5B C2 08 2C 69 00038	0002C 0002F 00032 00038	BGEQ 1\$ ADDL2 R11, R9 SUBL2 R11, R10 MOVCS #8, P.AAA, #0, R10, (R9)		
0060	8F	00	66 AE 08 AE 6E	020C	08 A1 00039 1\$: CE 9E 0003E	00039 0003E	ADDW3 #8, (R6), DESC MOVAB DEF_BUFF, DESC+4 MOVCS #0, (SP), #0, #96, \$RMS_PTR	0868 0870 0878	
			FF5C CD 6002	FF5C 6002	8F 8F 8E 01 8E 00055	0004B 0004E 00055	MOVW #24578, \$RMS_PTR MNEGB #1, \$RMS_PTR+2		
			FF5E CD FE04	FE04	9E 0005A	0005A	MOVAB SND_ESNAM_BUF, \$RMS_PTR+4 MNEGB #1, \$RMS_PTR+10		
			FF60 CD FE04	FE04	01 8E 00061	00061	MOVAB SND_ESNAM_BUF, \$RMS_PTR+12		
			FF66 CD FE04	FE04	9E 00066	00066	MOVZWL #518, -(SP)	0881	
			FF68 CD 7E	0204	8F 3C 0006D	0006D	PUSHL UIC	0880	
				OC AC DD 00072	9F 00075	00075	PUSHAB DESC		
				FF5C CD 9F 00078	9F 00078	00078	PUSHAB SND_NAM		
				BC AD 9F 0007C	9F 0007C	0007C	PUSHAB SND_RAB		
				56 10 AC DO 0007F	DO 0007F	0007F	MOVL SND_FAB, R6		
				00000000G	56 D0 00083	00083	PUSHL R6		
					06 FB 00085	00085	CALLS #6, MAIL\$CREATE_ISAM		
					59 D0 0008C	0008C	MOVL R0, STATUS		
					26 59 E9 0008F	0008F	BLBC STATUS, 3\$	0883	
					20 1D A6 91 00092	00092	CMPB 29(R6), #32	0884	
					20 12 00096	00096	BNEQ 3\$		
					BC AD 9F 00098	9F 00098	PUSHAB SND_RAB	0889	
					7E 0241 8F 3C 0009B	0009B	MOVZWL #577, -(SP)		
					00000000G	00 9F 000A0	PUSHAB MAIL\$SD_NEWMAIL		
						58 DD C00A6	PUSHL R8		
					F92F	CF 9F 000A8	PUSHAB GET_RECORD		
					F95A	CF 05 FB 000AC	CALLS #5, MAIL\$WRITE_MSG		
					03	50 E9 000B1	BLBC STATUS, 2\$		
						027C	BRW 26\$		
						04 000B7	RET		
						2\$: 08 A6 D0 000B8	MOVL 8(R6), SAVE_STS	0892	
						3\$: OC A6 D0 000BC	MOVL 12(R6), SAVE_STV	0893	
						02 A6 B5 000C0	TSTW 2(R6)	0894	
						09 13 000C3	BEQL 4\$		
						56 DD 000C5	PUSHL R6	0895	
						00000000G	CALLS #1, SYSS\$CLOSE		
						08 A6 53 D0 000CE	MOVL SAVE_STS, 8(R6)	0896	
						OC A6 52 D0 000D2	MOVL SAVE_STV, 12(R6)	0897	
						0001828A	CMPB STATUS, #98954	0901	
						8F 59 D1 000D6	BNEQ 5\$		
						20 1D A6 91 000DF	CMPB 29(R6), #32	0902	
						03 12 000E3	BNEQ 5\$		
						0001829A	BRW 11\$		
						8F 59 D1 000E8	CMPB STATUS, #98970	0905	
						7A 13 000EF	BEQL 9\$		
						1D A6 94 000F1	CLRB 29(R6)	0908	
						16 A6 11 B0 000F4	MOVW #17, 22(R6)	0909	
						1E A6 0202 8F B0 000F8	MOVW #514, 30(R6)	0910	

0058	8F	00	24	A6	FF04	CD	9E	000FE	MOVAB	SND_XABPRO, 36(R6)	0912	
				6E		0C	2C	00104	MOVCS	#0, (SP), #0, #88, \$RMS_PTR	0914	
					FF04	CD		0010B				
					5813	8F	B0	0010E	MOVW	#22547, \$RMS_PTR		
					OC	AC	D0	00115	MOVL	UIC, SND_XABPRO+12	0915	
						34	AE	0011B	MNEGW	#52, SND_XABPRO+8	0916	
						56	DD	00120	PUSHL	R6	0920	
		0000000G		00		01	FB	00122	CALLS	#1, SYSS\$CREATE		
				59		50	D0	00129	MOVL	R0, STATUS		
		0001828A		8F		59	D1	0012C	CMPL	STATUS, #98954	0921	
						10	12	00133	BNEQ	6\$		
				07	A6	02	8A	00135	BICB2	#2, 7(R6)	0923	
						56	DD	00139	PUSHL	R6	0924	
		0000000G		00		01	FB	0013B	CALLS	#1, SYSS\$CREATE		
				59		50	D0	00142	MOVL	R0, STATUS		
		04		40	A6	03	E1	00145	6\$:	BBC	#3, 64(R6), 7\$	0927
						17	A6	0014A	7\$:	BLBC	67(R6), 8\$	0928
						56	DD	0014E	7\$:	PUSHL	R6	0930
		0000000G		00		01	FB	00150	CALLS	#1, SYSS\$CLOSE		
				59	0144	8F	3C	00157	MOVZWL	#324, STATUS	0931	
				08	A6	8F	3C	0015C	MOVZWL	#324, 8(R6)	0932	
					OC	A6	D4	00162	CLRL	12(R6)	0933	
				OC	AC	CD	D1	00165	8\$:	CMPL	SND_XABPRO+12, UIC	0935
						07	13	0016B	9\$:	BEQL	10\$	
				59	0001829A	8F	D0	0016D	MOVW	#98970, STATUS	0936	
				0A		59	E8	00174	10\$:	BLBS	STATUS, 12\$	0938
						56	DD	00177	11\$:	PUSHL	R6	0939
		0000000G		00		01	FB	00179	CALLS	#1, UTIL\$REPORT_IO_ERROR		
						04		00180	RET			
0044	8F	00		6E		00	2C	00181	12\$:	MOVCS	#0, (SP), #0, #68, \$RMS_PTR	0944
					BC	AD		00188				
					4401	8F	B0	0018A	MOVW	#17409, \$RMS_PTR		
					0100	8F	3C	00190	MOVZWL	#256, \$RMS_PTR+4		
					007E10A2	8F	D0	00196	MOVL	#8261794, \$RMS_PTR+24		
						56	D0	0019E	MOVL	R6, \$RMS_PTR+60		
					0000000G	00	9F	001A2	PUSHAB	UTIL\$REPORT_IO_ERROR	0946	
					BC	AD	9F	001A8	PUSHAB	SND_RAB		
		0000000G		00		02	FB	001AB	CALLS	#2, SYSS\$CONNECT		
				59		50	D0	001B2	MOVL	R0, STATUS		
				04		59	E8	001B5	BLBS	STATUS, 13\$	0947	
				50		59	D0	001B8	MOVL	STATUS, R0	0948	
						04		001BB	RET			
					007E10D0	8F	D0	001BC	13\$:	MOVL	#8261840, SND_RAB+24	0949
		28		D4	AD	03	E1	001C4	BBC	#3, 5(R7), 14\$	0953	
				05	A7	57	DD	001C9	PUSHL	R7	0955	
						01	FB	001CB	CALLS	#1, SYSS\$DISCONNECT		
		0000000G		00		08	8A	001D2	BICB2	#8, 5(R7)	0956	
				05	A7	8F	B0	001D6	MOVW	#512, 32(R7)	0957	
				20	A7	C8	9E	001DC	MOVAB	134(R8), 36(R7)	0958	
				24	A7	00	9F	001E2	PUSHAB	UTIL\$REPORT_IO_ERROR	0959	
					0000000G	57	DD	001E8	PUSHL	R7		
		0000000G		00		02	FB	001EA	CALLS	#2, SYSS\$CONNECT		
					BC	AD	9F	001F1	14\$:	PUSHAB	SND_RAB	0965
					FDEB	CF	9F	001F4	PUSHAB	P.AXB		
						01	DD	001F8	PUSHL	#1		
						03	FB	001FA	CALLS	#3, PUT_SEND_MSG		
				F72D	CF	50	E9	001FF	BLBC	STATUS, 15\$		
					5F							

04	AE	0200	8F	60	00202	MOVW	#512, DESC	0969
08	AE	0C	AE	9E	00208	MOVAB	REC_BUFF, DESC+4	0970
			7E	D4	0020D	CLRL	-(SP)	0975
		08	AB	9F	0020F	PUSHAB	8(R8)	
		0C	AE	9F	00212	PUSHAB	DESC	
		0C	AE	9F	00215	PUSHAB	REC_SIZ	
		FDDC	CF	9F	00218	PUSHAB	P.AAC	
00000000G	00		05	FB	0021C	CALLS	#5, SYSS\$FAO	
		BC	AD	9F	00223	PUSHAB	SND_RAB	0977
		10	AE	9F	00226	PUSHAB	REC_BUFF	
		08	AE	DD	00229	PUSHL	REC_SIZ	
F6FB	CF		03	FB	0022C	CALLS	#3, PUT_SEND_MSG	
	72		50	E9	00231	BLBC	STATUS, -17\$	
		10	AB	9F	00234	PUSHAB	16(R8)	0982
		08	AE	9F	00237	PUSHAB	DESC	
		08	AE	9F	0023A	PUSHAB	REC_SIZ	
		007EDE10	8F	DD	0023D	PUSHL	#83T4384	
00000000G	00		01	FB	00243	CALLS	#1, MAIL\$GETMSG	
00000000G	00		50	DD	0024A	PUSHL	R0	
		BC	04	FB	0024C	CALLS	#4, SYSS\$FAO	
		10	AD	9F	00253	PUSHAB	SND_RAB	0984
		08	AE	9F	00256	PUSHAB	REC_BUFF	
F6CB	CF		03	FB	00259	PUSHL	REC_SIZ	
	42		50	E9	00261	CALLS	#3, PUT_SEND_MSG	
		18	AB	B5	00264	BLBC	STATUS, -17\$	0986
			30	13	00267	TSTW	24(R8)	
		18	AB	9F	00269	BEQL	16\$	0991
		08	AE	9F	0026C	PUSHAB	24(R8)	
		08	AE	9F	0026F	PUSHAB	DESC	
		007EDE18	8F	DD	00272	PUSHAB	REC_SIZ	
00000000G	00		01	FB	00278	PUSHL	#83T4392	
00000000G	00		50	DD	0027F	CALLS	#1, MAIL\$GETMSG	
		BC	04	FB	00281	PUSHL	R0	
		10	AD	9F	00288	CALLS	#4, SYSS\$FAO	
		08	AE	9F	0028B	PUSHAB	SND_RAB	0993
			AE	DD	0028E	PUSHAB	REC_BUFF	
F696	CF		03	FB	00291	PUSHL	REC_SIZ	
	54		50	E9	00296	CALLS	#3, PUT_SEND_MSG	
		BC	AD	9F	00299	BLBC	STATUS, -20\$	0997
		10	AE	9F	0029C	PUSHAB	SND_RAB	
			7E	D4	0029F	PUSHAB	REC_BUFF	
F686	CF		03	FB	002A1	CLRL	-(SP)	
	44		50	E9	002A6	CALLS	#3, PUT_SEND_MSG	
			54	D4	002A9	BLBC	STATUS, -20\$	1001
		CJ000000G	00	9F	002AB	CLRL	FF_FLAG	1007
			57	DD	002B1	PUSHAB	UTIL\$REPORT_IO_ERROR	
00000000G	00		02	FB	002B3	PUSHL	R7	
0001827A	0A		50	E8	002BA	CALLS	#2, SYSS\$GET	
	8F		50	D1	002BD	BLBS	STATUS, 19\$	
			6D	13	002C4	CMP	STATUS, #98938	
			04	002C6	BEQL	26\$		
	2A		54	E9	002C7	RET		
	52	22	A7	9E	002CA	BLBC	FF_FLAG, 22\$	1008
	53		62	3C	002CE	MOVAB	34(R7), R2	1013
	04		53	B1	002D1	MOVZWL	(R2), R3	
			1A	1B	002D4	CMPW	R3, #4	
						BLEQU	21\$	

6D6F7246	8F	28	B7	D1	002D6		CMPL	@40(R7), #1836020294	:	1014
			10	12	002DE		BNEQ	21\$:	
		BC	AD	9F	002E0		PUSHAB	SND RAB	:	1016
		28	A7	DD	002E3		PUSHL	40(R7)	:	
			7E	D4	002E6		CLRL	-(SP)	:	
F63F	CF		03	FB	002E8		CALLS	#3, PUT_SEND_MSG	:	
	46		50	E9	002ED	20\$:	BLBC	STATUS, 27\$:	
			54	D4	002F0	21\$:	CLRL	FF_FLAG	:	1017
			15	11	002F2		BRB	23\$:	1008
	52	22	A7	9E	002F4	22\$:	MOVAB	34(R7), R2	:	1019
	53		62	3C	002FB		MOVZWL	(R2), R3	:	
	01		53	B1	002FB		CMPW	R3, #1	:	
			09	12	002FE		BNEQ	23\$:	
	0C	28	B7	91	00300		CMPB	@40(R7), #12	:	1020
			03	12	00304		BNEQ	23\$:	
	54		01	D0	00306		MOVL	#1, FF_FLAG	:	1021
	02		53	B1	00309	23\$:	CMPW	R3, #2	:	1026
			10	1F	0030C		BLSSU	24\$:	
50	53	28	A7	C1	0030E		ADDL3	40(R7), R3, R0	:	1027
	8F	FE	A0	B1	00313		CMPW	-2(R0), #2573	:	1028
			03	12	00319		BNEQ	24\$:	
	62		02	A2	0031B		SUBW2	#2, (R2)	:	1029
		BC	AD	9F	0031E	24\$:	PUSHAB	SND RAB	:	1032
		28	A7	DD	00321		PUSHL	40(R7)	:	
			62	3C	00324		MOVZWL	(R2), -(SP)	:	
	7E		03	FB	00327		CALLS	#3, PUT_SEND_MSG	:	
F600	CF		50	E9	0032C		BLBC	STATUS, 25\$:	
	03		FF79	31	0032F		BRW	18\$:	
				04	00332	25\$:	RET		:	
	50		01	D0	00333	26\$:	MOVL	#1, R0	:	1036
				04	00336	27\$:	RET		:	1037

: Routine Size: 823 bytes, Routine Base: \$CODE\$ + 06D4

```

1038 1 GLOBAL ROUTINE MAIL$SEND_MSG(CNCTDESC,DEVDIR_DESC,UIC,USERDESC) =
1039 1 ++
1040 1 FUNCTIONAL DESCRIPTION:
1041 1
1042 1 This routine is called to send a message to another user. It opens
1043 1 the user's new mail message file and copies the new message to the
1044 1 file. Since there is no file sharing, if the new mail message file
1045 1 cannot be opened because it is already in use, another file will be
1046 1 created. The files will all be concatenated into one file when read.
1047 1
1048 1 INPUTS:
1049 1
1050 1 CNCTDESC = Address of CNCT block for message
1051 1 DEVDIR_DESC = Address of descriptor of addressee's mail directory
1052 1 UIC = OIC of addressee.
1053 1 USERDESC = Address of descriptor of user's name
1054 1
1055 1 OUTPUTS:
1056 1
1057 1 All failures are signalled as well as returned.
1058 1 --
1059 2 BEGIN
1060 2
1061 2 MAP
1062 2 CNCTDESC : REF $BBLOCK;
1063 2
1064 2 LOCAL
1065 2 SND_FAB : $FAB_DECL,
1066 2 STATUS;
1067 2
1068 2 $REWIND(RAB=CNCTDESC[CNCT T RAB]);
1069 2 MAIL$UP_PRIV(1*$BITPOSITION(PRV$V SYSPRV),0);
1070 2 STATUS = SEND_MSG(.CNCTDESC,.DEVDIR_DESC,.UIC,SND_FAB);
1071 2 SND_FAB[FAB$XAB] = 0;
1072 2 SND_FAB[FAB$CTX] = MAIL$CLOSEOUT;
1073 2 IF .SND_FAB[FAB$W !FI] NEQ 0
1074 2 THEN $CLOSE(FAB=SND_FAB,ERR=UTIL$REPORT_IO_ERROR);
1075 2 MAIL$DOWN_PRIV();
1076 2
1077 2 Increment new mail count
1078 2
1079 2 IF .STATUS
1080 2 THEN MAIL$INCR_NEWMAIL(.USERDESC);
1081 2
1082 2 RETURN .STATUS
1083 1 END;

```

```

                                .EXTRN  SYS$REWIND
                                .ENTRY  MAIL$SEND_MSG, Save R2
7E      04      SE      B0      AE      9E 00002
00000000G  00 00000286  8F      C1 00006
                                01  FB 0000F
                                7E  D4 00016
                                8F  DD 00018
                                MOVAB  -80(SP), SP
                                ADDL3  #646, CNCTDESC, -(SP)
                                CALLS  #1, SYS$REWIND
                                CLRL   -(SP)
                                PUSHL  #268435456

```

1038
1068
1069

00000000G	00		02	FB	0001E	CALLS	#2, MAIL\$UP_PRIV	:	
			5E	DD	00025	PUSHL	SP	:	1070
	7E	08	AC	7D	00027	MOVQ	DEVDIR_DESC, -(SP)	:	
		04	AC	DD	00028	PUSHL	CNCTDESC	:	
FC96	CF		04	FB	0002E	CALLS	#4, SEND_MSG	:	
	52		50	DO	00033	MOVL	R0, STATUS	:	
		24	AE	D4	00036	CLRL	SND_FAB+36	:	1071
18	AE	007E1058	8F	DO	00039	MOVL	#8281720, SND_FAB+24	:	1072
		02	AE	B5	00041	TSTW	SND_FAB+2	:	1073
			10	13	00044	BEQL	1\$:	
		00000000G	00	9F	00046	PUSHAB	UTIL\$REPORT_IO_ERROR	:	1074
		04	AE	9F	0004C	PUSHAB	SND_FAB	:	
00000000G	00		02	FB	0004F	CALLS	#2, SYSSCLOSE	:	
00000000G	00		00	FB	00056	CALLS	#0, MAIL\$DOWN_PRIV	:	1075
	0A		52	E9	0005D	BLBC	STATUS, 2\$:	1079
		10	AC	DD	00060	PUSHL	USERDESC	:	1080
00000000G	00		01	FB	00063	CALLS	#1, MAIL\$INCR_NEWMAIL	:	
	50		52	DO	0006A	MOVL	STATUS, R0	:	1082
			04	0006D	RET			:	1083

; Routine Size: 110 bytes, Routine Base: \$CODE\$ + 0A0B

```

950 1084 1 ROUTINE SEND_ERROR (SIGARG,MECHARG,ENBL) =
951 1085 1 |++
952 1086 1 | FUNCTIONAL DESCRIPTION:
953 1087 1 |
954 1088 1 |     Errors signalled by MAIL$SEND_ADDR and routines it calls come here
955 1089 1 |
956 1090 1 |--
957 1091 2 BEGIN
958 1092 2 MAP
959 1093 2     SIGARG : REF $BBLOCK,
960 1094 2     MECHARG : REF $BBLOCK,
961 1095 2     ENBL : REF VECTOR[.LONG];
962 1096 2
963 1097 2 END
964 1098 2     ADRDESC = .ENBL[1] : REF $BBLOCK;
965 1099 2
966 1100 2 LOCAL
967 1101 2     USERDESC : VECTOR[2,LONG],
968 1102 2     LNK : REF $BBLOCK,
969 1103 2     DESC : VECTOR[2,LONG];
970 1104 2
971 1105 2 IF NOT .MAIL$GL_FLAGS[MAIF_V_NETJOB]
972 1106 2 AND .ADRDESC NEQ 0
973 1107 3 THEN BEGIN
974 1108 3     USERDESC[0] = .ADRDESC[ADR_B_NAMLANG];
975 1109 3     USERDESC[1] = ADRDESC[ADR_T_NAME];
976 1110 3     IF (LNK = .ADRDESC[ADR_L_LNK]) NEQ 0
977 1111 4     THEN BEGIN
978 1112 4         DESC[0] = .LNK[LNK_B_NODLEN];
979 1113 4         DESC[1] = LNK[LNK_T_NODE];
980 1114 4         SIGNAL(MAIL$_SENDERRAT,2,USERDESC,DESC);
981 1115 4     END
982 1116 3     ELSE SIGNAL(MAIL$_SENDERR,1,USERDESC);
983 1117 2 END;
984 1118 2
985 1119 2 RETURN $$$_RESIGNAL
986 1120 1 END;

```

```

0004 00000 SEND_ERROR:
          52 00000000G 00 9E 00002      .WORD      Save R2
          5E          10 C2 00009      MOVAB      LIB$SIGNAL, R2
          50          0C AC D0 0000C      SUBL2     #16, SP
42 00000000G 00          01 E0 00010      MOVL      ENBL, R0
          04          B0 D5 00018      BBS       #1, MAIL$GL_FLAGS, 2$
          3D          13 0001B      TSTL     @4(R0)
          50          04 B0 D0 0001D      BEQL     2$
          08 AE          1D A0 9A 00021      MOVL     @4(R0), R0
          0C AE          1E A0 9E 00026      MOVZBL   29(R0), USERDESC
          50          08 A0 D0 0002B      MOVAB    30(R0), USERDESC+4
          1B          13 0002F      MOVL     8(R0), LNK
          6E          2F A0 9A 00031      BEQL     1$
          04 AE          30 A0 9E 00035      MOVZBL   47(LNK), DESC
          50          08 A0 D0 0002B      MOVAB    48(R0), DESC+4

```

```

: 1084
:
: 1098
: 1105
: 1106
:
: 1108
:
: 1109
: 1110
:
: 1112
: 1113

```

		SE	DD	0003A	PUSHL	SP	:	1114
	0C	AE	9F	0003C	PUSHAB	USERDESC	:	
		02	DD	0003F	PUSHL	#2	:	
62	007E810A	8F	DD	00041	PUSHL	#8290570	:	
		04	FB	00047	CALLS	#4, LIB\$SIGNAL	:	
		0E	11	0004A	BRB	2\$:	1110
	08	AE	9F	0004C	PUSHAB	USERDESC	:	1116
		01	DD	0004F	PUSHL	#1	:	
62	007E803A	8F	DD	00051	PUSHL	#8290362	:	
		03	FB	00057	CALLS	#3, LIB\$SIGNAL	:	
50	0918	8F	3C	0005A	MOVZWL	#2328, R0	:	1119
		04	0005F	RET			:	1120

; Routine Size: 96 bytes, Routine Base: \$CODE\$ + 0A79

```

: 988      1121 1 GLOBAL ROUTINE MAIL$SEND_ADDR (CNCTDESC) =
: 989      1122 1 |++
: 990      1123 1 | FUNCTIONAL DESCRIPTION:
: 991      1124 1 |
: 992      1125 1 |         Send the message to all addressees in the addressee list
: 993      1126 1 |
: 994      1127 1 | Inputs:
: 995      1128 1 |
: 996      1129 1 |         cnctdesc = address of cnct block for message to send.
: 997      1130 1 |--
: 998      1131 2 BEGIN
: 999      1132 2
: 1000     1133 2 MAP
: 1001     1134 2     CNCTDESC : REF $BBLOCK;
: 1002     1135 2
: 1003     1136 2 BIND
: 1004     1137 2     CNCTRAB = CNCTDESC[CNCT_T_RAB] : $BBLOCK,
: 1005     1138 2     CNCTFAB = .CNCTRAB[RAB$_FAB] : $BBLOCK;
: 1006     1139 2
: 1007     1140 2 BUILTIN
: 1008     1141 2     REMQUE;
: 1009     1142 2
: 1010     1143 2 LOCAL
: 1011     1144 2     ADRDESC : VOLATILE REF $BBLOCK,
: 1012     1145 2     USERDESC : VECTOR[2, LONG],
: 1013     1146 2     XABFHC : $XABFHC DECL,
: 1014     1147 2     TDESC : $BBLOCK[DSC$_S_BLN],
: 1015     1148 2     DFILE,
: 1016     1149 2     FIRST TIME,
: 1017     1150 2     STATUS;
: 1018     1151 2
: 1019     1152 2 |
: 1020     1153 2 |     ADRDESC will be cleared by ENABLE
: 1021     1154 2 |
: 1022     1155 2 | ENABLE SEND_ERROR (ADRDESC);
: 1023     1156 2 |
: 1024     1157 2 |
: 1025     1158 2 |     If not network job, or network and file received ok, then get file size.
: 1026     1159 2 |     Ensure it's a disk oriented device.  If not, assume the null device
: 1027     1160 2 |
: 1028     1161 2 | DFILE = .CNCTFAB[$FAB_DEV(MNT)] AND .CNCTFAB[$FAB_DEV(FOD)]
: 1029     1162 2 |         AND .CNCTFAB[$FAB_DEV(RND)];
: 1030     1163 2 | CNCTRAB[RAB$_CTX] = MAIL$READERR;
: 1031     1164 2 | CNCTDESC[CNCT_L_MSGBLKS] = 0;
: 1032     1165 2 |
: 1033     1166 2 | IF (NOT .MAIL$GL_FLAGS[MAIF V NETJOB]
: 1034     1167 2 |     OR .CNCTDESC[CNCT_L_FILEERR])
: 1035     1168 2 |     AND .DFILE
: 1036     1169 2 | THEN BEGIN
: 1037     1170 2 |     $REWIND(RAB=CNCTRAB, ERR=UTIL$REPORT_IO_ERROR);
: 1038     1171 2 |     $XABFHC_INIT(XAB=XABFHC);
: 1039     1172 2 |     CNCTFAB[FAB$_XAB] = XABFHC;
: 1040     1173 2 |     $DISPLAY(FAB=CNCTFAB, ERR=UTIL$REPORT_IO_ERROR);
: 1041     1174 2 |     CNCTFAB[FAB$_XAB] = 0;
: 1042     1175 2 |     CNCTDESC[CNCT_L_MSGBLKS] = .XABFHC[XAB$_EBK];
: 1043     1176 2 |     IF .XABFHC[XAB$_FFB] EQL 0
: 1044     1177 2 |         THEN CNCTDESC[CNCT_L_MSGBLKS] = .CNCTDESC[CNCT_L_MSGBLKS] - 1;

```



```
1045 1178 2 END;
1046 1179 2
1047 1180 2 Set up the correct fromdesc in the cnct block
1048 1181 2
1049 1182 2 IF .MAIL$Q_SYSNODE2[DSC$W_LENGTH] NEQ 0
1050 1183 2 AND NOT .MAIL$GL_FLAGS[MAIF_V_NETJOB]
1051 1184 2 THEN STR$CONCAT(DYN_DESC_2,MAIL$Q_SYSNODE2,CNCTDESC[CNCT_Q_FROMDESC])
1052 1185 2 ELSE STR$COPY_DX(DYN_DESC_2,CNCTDESC[CNCT_Q_FROMDESC]);
1053 1186 2
1054 1187 2 Loop through all addressees.
1055 1188 2
1056 1189 2 FIRST_TIME = TRUE;
1057 1190 2 WHILE NOT REMQUE(. (CNCTDESC[CNCT_Q_ADRLIST]),ADRDESC)
1058 1191 2 DO IF NOT .ADRDESC[ADR_V_NOSEND]
1059 1192 2 THEN BEGIN
1060 1193 2 USERDESC[0] = .ADRDESC[ADR_B_NAMING];
1061 1194 2 USERDESC[1] = ADRDESC[ADR_T_NAME];
1062 1195 2
1063 1196 2 If network job and file error encountered, report it now,
1064 1197 2 otherwise, attempt to deliver the mail
1065 1198 2
1066 1199 2 IF .MAIL$GL_FLAGS[MAIF_V_NETJOB]
1067 1200 2 AND NOT .CNCTDESC[CNCT_L_FILEERR]
1068 1201 2 THEN SIGNAL(.CNCTDESC[CNCT_L_FILEERR],1,UTIL$GETFILENAME(CNCTFAB),
1069 1202 2 .CNCTDESC[CNCT_L_FILESTS],.CNCTDESC[CNCT_L_FILESTV])
1070 1203 2 ELSE BEGIN
1071 1204 2 IF NOT .FIRST_TIME
1072 1205 2 AND .DFILE
1073 1206 2 THEN $REWIND(RAB=CNCTRAB);
1074 1207 2 FIRST_TIME = FALSE;
1075 1208 2 IF .ADRDESC[ADR_L_LLNK] NEQ 0
1076 1209 2 THEN BEGIN
1077 1210 2 STATUS = MAIL$NET_SEND(.ADRDESC,.CNCTDESC);
1078 1211 2 IF .STATUS
1079 1212 2 AND .MAIL$GL_FLAGS[MAIF_V_NETJOB]
1080 1213 2 THEN SIGNAL(T);
1081 1214 2 END
1082 1215 2 ELSE BEGIN
1083 1216 2 IF MAIL$GET_DEVDIR(USERDESC,DYN_DESC)
1084 1217 2 THEN BEGIN
1085 1218 2 IF NOT .MAIL$GL_FLAGS[MAIF_V_NETJOB]
1086 1219 2 THEN BEGIN
1087 1220 2 CH$MOVE(DSC$S_BLN,CNCTDESC[CNCT_Q_FROMDESC],TDESC);
1088 1221 2 CH$MOVE(DSC$S_BLN,DYN_DESC_2,CNCTDESC[CNCT_Q_FROMDESC]);
1089 1222 2 END;
1090 1223 2 STATUS = MAIL$SEND_MSG(.CNCTDESC,DYN_DESC,
1091 1224 2 .MAIL$T_UAF_BUFF[UAF$UIC],USERDESC);
1092 1225 2 IF NOT .MAIL$GL_FLAGS[MAIF_V_NETJOB]
1093 1226 2 THEN CH$MOVE(DSC$S_BLN,TDESC,CNCTDESC[CNCT_Q_FROMDESC]);
1094 1227 2
1095 1228 2 Update new mail count
1096 1229 2
1097 1230 2 IF .STATUS
1098 1231 2 THEN BEGIN
1099 1232 2 MAIL$NOTIFY(USERDESC,CNCTDESC[CNCT_Q_FROMDESC]);
1100 1233 2 IF .MAIL$GL_FLAGS[MAIF_V_NETJOB]
1101 1234 2 AND .STATUS
```

: 1102
: 1103
: 1104
: 1105
: 1106
: 1107
: 1108
: 1109
: 1110
: 1111
: 1112
: 1113

1235 7
1236 6
1237 5
1238 4
1239 3
1240 2
1241 1
1242
1243
1244
1245
1246

```
THEN SIGNAL(1);  
END;  
END;  
END;  
END;  
Deallocate the addressee List entry  
LIB$FREE_VM(%REF(.ADRDESC[ADR_B_NAMLANG]+ADR_C_LENGTH),ADRDESC);  
END;  
RETURN TRUE  
END;
```

OFFC	00000	.EXTRN	SYSS\$DISPLAY	.ENTRY	MAIL\$SEND ADDR, Save R2,R3,R4,R5,R6,R7,R8,- R9,R10,R11	
					-68(SP), SP	: 1121
				MOVAB	CNCTDESC, R6	: 1137
				MOVAB	646(R6), R8	: 1138
				MOVBL	60(R8), R7	: 1161
				CLRL	ADRDESC	: 1162
				MOVAL	16\$, (FP)	: 1163
				EXTZV	#3, #1, 66(R7), R0	: 1164
				EXTZV	#6, #1, 65(R7), R1	: 1166
				MCOML	R1, R1	: 1167
				BICL2	R1, R0	: 1168
				EXTZV	#4, #1, 67(R7), DFILE	: 1170
				MCOML	DFILE, DFILE	: 1171
				BICL3	DFILE, R0, DFILE	: 1172
				MOVL	#8261810, 24(R8)	: 1173
				CLRL	60(R6)	: 1174
				BBC	#1, MAIL\$GL_FLAGS, 1\$: 1175
				BLBC	72(R6), 2\$: 1176
				BLBC	DFILE, 2\$: 1177
				PUSHAB	UTIL\$REPORT_IO_ERROR	: 1184
				PUSHL	R8	: 1182
				CALLS	#2, SYSS\$REWIND	: 1183
				MOVCS	#0, (SP), #0, #44, \$RMS_PTR	: 1184
				MOVW	#11293, \$RMS_PTR	: 1174
				MOVAB	XABFHC, 36(R7)	: 1175
				PUSHAB	UTIL\$REPORT_IO_ERROR	: 1176
				PUSHL	R7	: 1177
				CALLS	#2, SYSS\$DISPLAY	: 1184
				CLRL	36(R7)	: 1182
				MOVL	XABFHC+16, 60(R6)	: 1183
				TSTW	XABFHC+20	: 1184
				BNEQ	2\$: 1177
				DECL	60(R6)	: 1184
				ADDL2	#8, R6	: 1182
				TSTW	MAIL\$Q_SYSDNODE2	: 1183
				BEQL	3\$: 1184
				BBS	#1, MAIL\$GL_FLAGS, 3\$: 1183
				PUSHL	R6	: 1184

		00000000G	00	9F	000A9	PUSHAB	MAIL\$Q SYSNODE2			
		00000000'	00	9F	000AF	PUSHAB	DYN_DESC 2			
	00000000G	00	03	FB	000B5	CALLS	#3, -STR\$CONCAT			
			0F	11	000BC	BRB	4\$			
			56	DD	000BE	3\$: PUSHL	R6	1185		
		00000000'	00	9F	000C0	PUSHAB	DYN_DESC 2			
	00000000G	00	02	FB	000C6	CALLS	#2, -STR\$COPY DX			
			01	DO	000CD	4\$: MOVL	#1, FIRST_TIME	1189		
			04	AC	000D0	5\$: MOVL	CNCTDESC, -R0	1190		
	40	AE	28	BO	000D4	REMQUE	@40(R0), ADRDESC			
			03	1C	000D9	BVC	6\$			
			011E	31	000DB	BRW	15\$			
			40	AE	000DE	6\$: MOVL	ADRDESC, R0	1191		
			1C	A0	000E2	BLBS	28(R0), 5\$			
			40	AE	000E6	MOVL	ADRDESC, R0	1193		
	38	AE	1D	A0	000EA	MOVZBL	29(R0), USERDESC			
3C	AE	40	AE	1E	000EF	ADDL3	#30, ADRDESC, USERDESC+4	1194		
	2A	00000000G	00	01	E1	000F5	BBC	#1, MAIL\$GL_FLAGS, 8\$	1199	
			50	AC	000FD	MOVL	CNCTDESC, R0	1200		
			22	A0	00101	BLBS	72(R0), 8\$			
			52	AC	001J5	MOVL	CNCTDESC, R2	1202		
			7E	4C	A2	7D	00109	MOVQ	76(R2), -(SP)	
				57	DD	0010D	PUSHL	R7	1201	
	00000000G	00	01	FB	0010F	CALLS	#1, UTIL\$GETFILENAME			
			50	DD	00116	PUSHL	R0			
			01	DD	00118	PUSHL	#1			
			48	A2	DD	0011A	PUSHL	72(R2)		
	00000000G	00	05	FB	0011D	CALLS	#5, LIB\$SIGNAL			
			00B8	31	00124	7\$: BRW	14\$			
			0C	59	E8	00127	8\$: BLBS	FIRST_TIME, 9\$	1204	
			09	5B	E9	0012A	BLBC	DFILE, 9\$	1205	
				58	DD	0012D	PUSHL	R8	1206	
	00000000G	00	01	FB	0012F	CALLS	#1, SYSS\$REWIND			
			59	D4	00136	9\$: CLRL	FIRST TIME	1207		
			50	40	AE	00138	MOVL	ADRDESC, R0	1208	
				08	A0	D5	0013C	TSTL	8(R0)	
				1D	13	0013F	BEQL	10\$		
			04	AC	DD	00141	PUSHL	CNCTDESC	1210	
			44	AE	DD	00144	PUSHL	ADRDESC		
	00000000G	00	02	FB	00147	CALLS	#2, MAIL\$NET_SEND			
			5A	50	DO	0014E	MOVL	R0, STATUS		
			DO	5A	E9	00151	BLBC	STATUS, 7\$	1211	
	C8	00000000G	00	01	E1	00154	BBC	#1, MAIL\$GL_FLAGS, 7\$	1212	
				78	11	0015C	BRB	13\$	1213	
				00	9F	0015E	10\$: PUSHAB	DYN_DESC	1216	
			3C	AE	9F	00164	PUSHAB	USERDESC		
	00000000G	00	02	FB	00167	CALLS	#2, MAIL\$GET_DEVDIR			
			6E	50	E9	0016E	BLBC	R0, 14\$		
	13	00000000G	00	01	E0	00171	BBS	#1, MAIL\$GL_FLAGS, 11\$	1218	
			56	04	AC	00179	MOVL	CNCTDESC, R6	1220	
	04	AE	08	A6	08	28	0017D	MOVC3	#8, 8(R6), TDESC	
08	A6	00000000'	00	08	28	00183	MOVC3	#8, DYN_DESC_2, 8(R6)	1221	
				38	AE	9F	0018C	11\$: PUSHAB	USERDESC	1223
			50	00000000G	00	DO	0018F	MOVL	MAIL\$T_UAF_BUFF, R0	1224
				24	A0	DD	00196	PUSHL	36(R0)	
				00000000'	00	9F	00199	PUSHAB	DYN_DESC	1223
			56	04	AC	DO	0019F	MOVL	CNCTDESC, R6	

	FD88	CF		56	DD	001A3		PUSHL	R6		
		5A		04	FB	001A5		CALLS	#4, MAIL\$SEND_MSG		
		00		50	DO	001AA		MOVL	R0, STATUS		
08	06	00000000G	00	01	E0	001AD		BBS	#1, MAIL\$GL_FLAGS, 12\$		1225
	A6	04	AE	08	28	001B5		MOVLC3	#8, TDESC, 8(R6)		1226
			21	5A	E9	001BB	12\$:	BLBC	STATUS, 14\$		1230
				08	A6	9F	001BE	PUSHAB	8(R6)		1232
				3C	AE	9F	001C1	PUSHAB	USERDESC		
	00000000G	00		02	FB	001C4		CALLS	#2, MAIL\$NOTIFY		
0C	00000000G	00		01	E1	001CB		BBC	#1, MAIL\$GL_FLAGS, 14\$		1233
		09		5A	E9	001D3		BLBC	STATUS, 14\$		1234
	00000000G	00		01	DD	001D6	13\$:	PUSHL	#1		1235
				01	FB	001D8		CALLS	#1, LIB\$SIGNAL		
				40	AE	9F	001DF	14\$:	PUSHAB	ADRDESC	1243
				44	AE	DO	001E2	MOVL	ADRDESC, R0		
	04	AE		1D	A0	9A	001E6	MOVZBL	29(R0), 4(SP)		
	04	AE			1E	C0	001EB	ADDL2	#30, 4(SP)		
	00000000G	00		04	AE	9F	001EF	PUSHAB	4(SP)		
					02	FB	001F2	CALLS	#2, LIB\$FREE_VM		
				FED4	31	001F9		BRW	5\$		1191
					01	DO	001FC	15\$:	MOVL	#1, R0	1245
					04	001FF		RET			1246
					0000	00200	16\$:	.WORD	Save nothing		1138
				08	AC	DO	00202	MOVL	8(AP), R0		
				04	A0	DO	00206	MOVL	4(R0), R0		
				FC	A0	9F	0020A	PUSHAB	ADRDESC		
					01	DD	0020D	PUSHL	#1		
					5E	DD	0020F	PUSHL	SP		
	FD86	7E		04	AC	7D	00211	MOVQ	4(AP), -(SP)		
		CF			03	FB	00215	CALLS	#3, SEND_ERROR		
					04	0021A		RET			

: Routine Size: 539 bytes, Routine Base: \$CODE\$ + 0AD9

: 1115 1247 1 END
: 1116 1248 0 ELUDOM

.EXTRN LIB\$SIGNAL

PSECT SUMMARY

Name	Bytes	Attributes
\$OWNS	16	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$CODES	3316	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	Symbols		Pages Mapped	Processing Time
	Total	Loaded Percent		
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	183 0	1000	00:01.4
_\$255\$DUA28:[MAIL.OBJ]MAILDEF.L32;1	457	52 11	26	00:00.4

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:SENDMSG/OBJ=OBJ\$:SENDMSG MSRC\$:SENDMSG/UPDATE=(ENH\$:SENDMSG)

: Size: 3276 code + 56 data bytes
: Run Time: 00:45.9
: Elapsed Time: 02:54.6
: Lines/CPU Min: 1630
: Lexemes/CPU-Min: 38945
: Memory Used: 486 pages
: Compilation Complete

