



```

SSSSSSSS YY YY SSSSSSSS MM MM AAAAAA IIIIII
SSSSSSSS YY YY SSSSSSSS MM MM AAAAAA IIIIII
SS SS YY YY SS SS MM MM AA AA II
SS SS YY YY SS SS MM MM AA AA II
SS SS YY YY SS SS MM MM AA AA II
SSSSSS YY YY SSSSSS SS SS MM MM AAAAAAAA II
SSSSSS YY YY SSSSSS SS SS MM MM AAAAAAAA II
SS SS YY YY SS SS MM MM AA AA II
SSSSSS YY YY SSSSSSSS MM MM AA AA IIIIII
SSSSSS YY YY SSSSSSSS MM MM AA AA IIIIII

```

```

LL IIIIII SSSSSSSS
LL IIIIII SSSSSSSS
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 IIIIII SSSSSSSS
LLLLLLLLLL IIIIII SSSSSSSS
LLLLLLLLLL IIIIII SSSSSSSS

```

```

1 0001 0 MODULE MAIL$SYSMAI( %TITLE 'OPEN/READ/WRITE SYS$SYSTEM:VMSMAIL.DAT'
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: This module contains routines that deal with the
34 0034 1 VMS MAIL data file, SYS$SYSTEM:VMSMAIL.DAT
35 0035 1
36 0036 1 ENVIRONMENT: NATIVE/USER MODE
37 0037 1
38 0038 1 AUTHOR: Benn Schreiber, CREATION DATE: 13-Dec-1982
39 0039 1
40 0040 1 MODIFIED BY:
41 0041 1
42 0042 1 V03-008 ROP0033 Robert Posniak 09-Aug-1984
43 0043 1 Make sure locally allocated XABs are disconnected
44 0044 1 from global FABs.
45 0045 1
46 0046 1 V03-007 BLS0255 Benn Schreiber 28-Dec-1983
47 0047 1 New global flags
48 0048 1
49 0049 1 V03-006 BLS0246 Benn Schreiber 7-Nov-1983
50 0050 1 Move new mail count into SYS$SYSTEM:VMSMAIL.DAT
51 0051 1
52 0052 1 V03-005 BLS0225 Benn Schreiber 17-Jun-1983
53 0053 1 Break up file spec into fns='VMSMAIL', dns='SYS$SYSTEM:.DAT',
54 0054 1 so that a system logical name for VMSMAIL can be used.
55 0055 1
56 0056 1 V03-004 BLS0221 Benn Schreiber 5-May-1983
57 0057 1 Correct SYSPRV handling

```

MAIL\$SYSMAI  
V04-000

OPEN/READ/WRITE SYS\$SYSTEM:VMSMAIL.DAT

N 14  
16-Sep-1984 01:23:17  
14-Sep-1984 12:42:35

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[MAIL.SRC]SYSMAI.B32;1 Page 2 (1)

: 58 0058 1 |  
: 59 0059 1 |  
: 60 0060 1 |  
: 61 0061 1 |  
: 62 0062 1 |  
: 63 0063 1 |  
: 64 0064 1 |--

V03-003 BLS0220 Benn Schreiber 30-Apr-1983  
Add MAIL\$GET\_DEV\_DIR  
V03-002 BLS0213 Benn Schreiber 21-Mar-1983  
Don't report open error if just reading VMSMAIL

MA  
VO

```
66 0065 1 |
67 0066 1 | TABLE OF CONTENTS
68 0067 1 |
69 0068 1 LIBRARY 'SYS$LIBRARY:LIB';
70 0069 1 REQUIRE 'SRC$MAILREQ';
71 0215 1 LIBRARY 'LIB$MAILDEF';
72 0216 1
73 0217 1 GLOBAL
74 0218 1 MAIL$L_SYSMAIBUF: REF $BLOCK, | Pointer to VMSMAIL.DAT buffer
75 0219 1 MAIL$L_USR$SYSBUF: REF $BLOCK, | Pointer to this user's
76 0220 1 | vmsmail record
77 0221 1 MAIL$T_SYSMAIFAB: $FAB_DECL, | FAB for I/O
78 0222 1 MAIL$T_SYSMAIRAB: $RAB_DECL; | and a RAB
79 0223 1
80 0224 1
81 0225 1 BIND
82 0226 1 VMSMAIL_DESC = $DESCRIPTOR('VMSMAIL') : $BLOCK,
83 0227 1 SYSEXE_DESC = $DESCRIPTOR('SYS$SYSTEM:.DAT') : $BLOCK;
84 0228 1
85 0229 1 OWN
86 0230 1 SYSMAI$ESS: $BLOCK[NAM$C_MAXRSS], | ESS for VMSMAIL
87 0231 1 SYSMAINAM: $NAM_DECL; | NAM block for VMSMAIL
88 0232 1
89 0233 1 EXTERNAL
90 0234 1 MAIL$T_UAF_BUFF : REF $BLOCK, | UAF buffer
91 0235 1 MAIL$Q_USERNAME: $BLOCK, | String descriptor of username
92 0236 1 MAIL$G_FLAGS : $BLOCK; | Control flags
93 0237 1
94 0238 1 EXTERNAL ROUTINE
95 0239 1 LIB$GET_VM: ADDRESSING_MODE(GENERAL), | Allocate virtual memory
96 0240 1 MAIL$GET_USER, | Get UAF record
97 0241 1 MAIL$UP_PRIV, | Raise privileges
98 0242 1 MAIL$DOWN_PRIV, | and drop them
99 0243 1 STR$APPEND, | String append
100 0244 1 STR$COPY_DX, | String copy
101 0245 1 UTIL$REPORT_IO_ERROR; | Report I/O error
102 0246 1 |
103 0247 1 | Define shared messages
104 0248 1 |
105 P 0249 1 $SHR_MSGDEF(MAIL,126,LOCAL,
106 P 0250 1 (READERR,WARNING),
107 P 0251 1 (WRITEERR,WARNING),
108 0252 1 (OPENOUT,WARNING));
```

```

110 0253 1 GLOBAL ROUTINE MAIL$UPDATE_SYSMAI_REC (RECBUF) =
111 0254 1 |++
112 0255 1 | FUNCTIONAL DESCRIPTION:
113 0256 1 |
114 0257 1 |     Write/update a new record in VMSMAIL for a user
115 0258 1 |
116 0259 1 | Inputs:
117 0260 1 |
118 0261 1 |     RECBUF = Address of record to write/update
119 0262 1 |
120 0263 1 | --
121 0264 2 BEGIN
122 0265 2 MAP
123 0266 2     RECBUF : REF $BLOCK;
124 0267 2
125 0268 2 LOCAL
126 0269 2     STATUS;
127 0270 2
128 0271 2 MAIL$T_SYSMAIRAB[RAB$W_RSZ] = VMD C LENGTH + .RECBUF[VMD B FWDLNG]
129 0272 2     + .RECBUF[VMD B FNMLNG] + .RECBUF[VMD B DIRLNG];
130 0273 2 MAIL$T_SYSMAIRAB[RAB$L_RBF] = .RECBUF[VMD B DIRLNG];
131 0274 2 MAIL$T_SYSMAIRAB[RAB$V_UIF] = 1;
132 0275 2 MAIL$T_SYSMAIRAB[RAB$L_KBF] = .MAIL$T_SYSMAIRAB[RAB$L_RBF];
133 0276 2 MAIL$T_SYSMAIRAB[RAB$L_CTX] = MAIL$ WRITEERR;
134 0277 2 STATUS = $PUT(RAB=MAIL$T_SYSMAIRAB,ERR=UTIL$REPORT_IO_ERROR);
135 0278 2 MAIL$T_SYSMAIRAB[RAB$V_UIF] = 0;
136 0279 2 IF .STATUS
137 0280 2     THEN STATUS = $FLUSH(RAB=MAIL$T_SYSMAIRAB,ERR=UTIL$REPORT_IO_ERROR);
138 0281 2
139 0282 2 | If this is the user's record, save the latest copy
140 0283 2 |
141 0284 2 IF .STATUS
142 0285 2     AND CH$EQL(VMD C USERNAME,RECBUF[VMD T USERNAME],
143 0286 2         VMD C USERNAME,MAIL$L_USR$SYSBUF[VMD T_USERNAME],%C' ')
144 0287 2     THEN CH$MOVE(.MAIL$T_SYSMAIRAB[RAB$W_RSZ],
145 0288 2         .MAIL$T_SYSMAIRAB[RAB$L_RBF],
146 0289 2         .MAIL$L_USR$SYSBUF);
147 0290 2
148 0291 2 RETURN .STATUS
149 0292 1 END;

```

.TITLE MAILSSYSMAI OPEN/READ/WRITE SYSS\$SYSTEM:VMSMAIL.DAT

.IDENT \V04-000\

.PSECT \$CODE\$,NOWRT,2

4C	49	41	4D	53	4D	56	00000	P.AAB:	.ASCII	\VMSMAIL\	:									
							00007		.BLKB	1	:									
							00000007	P.AAA:	.LONG	7	:									
							00000000'		.ADDRESS	P.AAB	:									
54	41	44	2E	3A	4D	45	54	53	59	53	24	53	59	53	00010	P.AAD:	.ASCII	\SYSS\$SYSTEM:.DAT\	:	
																0001F		.BLKB	1	:
																0000000F	P.AAC:	.LONG	15	:
																00000000'		.ADDRESS	P.AAD	:
																00024				:

.PSECT \$OWNS,NOEXE,2

00000 SYMAIESS:  
          .BLKB 255  
000FF       .BLKB 1  
00100 SYSMANAM:  
          .BLKB 96

.PSECT \$GLOBALS,NOEXE,2

00000 MAIL\$\_SYSMAIBUF::  
          .BLKB 4  
00004 MAIL\$\_USRSYSBUF::  
          .BLKB 4  
00008 MAIL\$\_SYSMAIFAB::  
          .BLKB 80  
00058 MAIL\$\_SYSMAIRAB::  
          .BLKB 68

VMSMAIL\_DESC= P.AAA  
SYSEXE\_DESC= P.AAC  
.EXTRN MAIL\$\_UAF\_BUFF  
.EXTRN MAIL\$\_USERNAME  
.EXTRN MAIL\$\_GC\_FLAGS, LIB\$GET\_VM  
.EXTRN MAIL\$\_GET\_USER, MAIL\$\_PRIV  
.EXTRN MAIL\$\_DOWN\_PRIV, STR\$APPEND  
.EXTRN STR\$\_COPY\_DX, UTIL\$\_REPORT\_IO\_ERROR  
.EXTRN SY\$\_PUT, SY\$\_FLUSH

.PSECT \$CODES,NOWRT,2

			01FC	00000	.ENTRY	MAIL\$UPDATE_SYSMAI_REC, Save R2,R3,R4,R5,-	: 0253
						R6,R7,R8	
	58	00000000G	00	9E	00002	MOVAB	UTIL\$REPORT IO ERROR, R8
	57	00000000'	00	9E	00009	MOVAB	MAIL\$_SYSMAIRAB+40, R7
	52	04	AC	DO	00010	MOVL	RECBUF, R2
	50	43	A2	9A	00014	MOVZBL	67(R2), R0
	51	42	A2	9A	00018	MOVZBL	66(R2), R1
	50		51	CO	0001C	ADDL2	R1, R0
	51	41	A2	9A	0001F	MOVZBL	65(R2), R1
	53	44	A140	9E	00023	MOVAB	68(R1)[R0], R3
FA	A7		53	BO	00028	MOVW	R3, MAIL\$_SYSMAIRAB+34
	67		52	DO	0002C	MOVL	R2, MAIL\$_SYSMAIRAB+40
DC	A7		10	88	0002F	BISB2	#16, MAIL\$_SYSMAIRAB+4
08	A7		67	DO	00033	MOVL	MAIL\$_SYSMAIRAB+40, MAIL\$_SYSMAIRAB+48
FO	A7	007E10D0	8F	DO	00037	MOVL	#8261840, MAIL\$_SYSMAIRAB+24
			58	DD	0003F	PUSHL	R8
			D8	A7	9F	PUSHAB	MAIL\$_SYSMAIRAB
00000000G	00		02	FB	00044	CALLS	#2, SY\$_PUT
	56		50	DO	0004B	MOVL	R0, STATUS
DC	A7		10	8A	0004E	BICB2	#16, MAIL\$_SYSMAIRAB+4
	28		56	E9	00052	BLBC	STATUS, 1\$
			58	DD	00055	PUSHL	R8
			D8	A7	9F	PUSHAB	MAIL\$_SYSMAIRAB
00000000G	00		02	FB	0005A	CALLS	#2, SY\$_FLUSH
	56		50	DO	00061	MOVL	R0, STATUS
	16		56	E9	00064	BLBC	STATUS, 1\$
							: 0271
							: 0272
							: 0273
							: 0274
							: 0275
							: 0276
							: 0277
							: 0278
							: 0279
							: 0280
							: 0284

MAIL\$SYSMAI  
V04-000

OPEN/READ/WRITE SYSS\$SYSTEM:VMSMAIL.DAT

E 15  
16-Sep-1984 01:23:17  
14-Sep-1984 12:42:35

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[MAIL.SRC]SYSMAI.B32;1 Page 6  
(3)

MA  
VO

60	50	84	A7	D0	00067	MOVL	MAIL\$L_USRSYSBUF, R0	:	0286
	62		1F	29	00068	CMPC3	#31, (R2), (R0)	:	
			0C	12	0006F	BNEQ	1\$	:	
	51		67	D0	00071	MOVL	MAIL\$T_SYSMAIRAB+40, R1	:	0288
	50	84	A7	D0	00074	MOVL	MAIL\$L_USRSYSBUF, R0	:	0289
60	61	FA	A7	28	00078	MOVCL	MAIL\$T_SYSMAIRAB+34, (R1), (R0)	:	
	50		56	D0	0007D	MOVL	STATUS, R0	:	0291
			04	00080	1\$: RET			:	0292

: Routine Size: 129 bytes, Routine Base: \$CODE\$ + 0028



```

: 151      0293 1 GLOBAL ROUTINE MAIL$OPEN_SYSMAI =
152      0294 2 BEGIN
153      0295 222 +-
154      0296 222 FUNCTIONAL DESCRIPTION:
155      0297 222
156      0298 222     Open SYS$SYSTEM:VMSMAIL.DAT for update.
157      0299 222
158      0300 222 INPUTS:
159      0301 222
160      0302 222     Static FAB/RAB
161      0303 222
162      0304 222 OUTPUTS:
163      0305 222
164      0306 222     file opened.
165      0307 222     Buffer allocated, pointed to by MAIL$L_SYSMAIBUF
166      0308 222
167      0309 222 ROUTINE VALUE:
168      0310 222
169      0311 222     success if file opened
170      0312 222     fail if not
171      0313 222
172      0314 222 --
173      0315 222
174      0316 222 LOCAL
175      0317 222     STATUS,
176      0318 222     XABPRO: $XABPRO_DECL,
177      0319 222     XABKEY: $XABKEY_DECL;
178      0320 222
179      0321 222     Allocate a buffer to read/write from
180      0322 222
181      0323 222 IF .MAIL$L_SYSMAIBUF EQL 0
182      P 0324 222     THEN IF_ERR(LIB$GET_VM(%REF(VMD C LENGTH+3*NAM$C_MAXRSS),
183      P 0325 222     MAIL$L_SYSMAIBUF),
184      0326 222     SIGNAL(.STATUS); RETURN TRUE);
185      0327 222 IF .MAIL$L_USR$SYSBUF EQL 0
186      P 0328 222     THEN IF_ERR(LIB$GET_VM(%REF(VMD C LENGTH+3*NAM$C_MAXRSS),
187      P 0329 222     MAIL$L_USR$SYSBUF),
188      0330 222     SIGNAL(.STATUS); RETURN TRUE);
189      0331 222
190      0332 222     Initialize RMS data structures
191      0333 222
192      P 0334 222 $FAB_INIT(FAB=MAIL$T_SYSMAIFAB,
193      P 0335 222     FAC=(GET,PUT,DEL,UPD),
194      P 0336 222     FOP=(CIF,CBT),
195      P 0337 222     FNS=.VMSMAIL_DESC[DSC$W_LENGTH],
196      P 0338 222     FNA=.VMSMAIL_DESC[DSC$A_POINTER],
197      P 0339 222     DNS=.SYSEXE_DESC[DSC$W_LENGTH],
198      P 0340 222     DNA=.SYSEXE_DESC[DSC$A_POINTER],
199      P 0341 222     SHR=(GET,PUT,DEL,UPD),
200      P 0342 222     CTX=MAIL$_OPENOUT,
201      P 0343 222     ORG=IDX,               !Indexed
202      P 0344 222     RFM=VAR,               !Variable length records
203      P 0345 222     ALQ=10,
204      0346 222     DEQ=10);
205      0347 222
206      0348 222 MAIL$T_SYSMAIFAB[FAB$V_LNM_MODE] = PSL$C_EXEC;
207      0349 222

```

```

: 208 P 0350 2 $NAM_INIT(NAM=SYSMAINAM,
: 209 P 0351 2 ESS=NAM$C_MAXRSS,
: 210 P 0352 2 ESA=SYSMAI$ESS,
: 211 P 0353 2 RSS=NAM$C_MAXRSS,
: 212 0354 2 RSA=SYSMAI$ESS);
: 213 0355 2
: 214 0356 2
: 215 P 0357 2 $XABPRO_INIT(XAB=XABPRO,
: 216 P 0358 2 PRO=(RWE,RWE), !Owner and system access only
: 217 P 0359 2 UIC=(1,4), !Owner is [1,4]
: 218 0360 2 NXT=XAB$KEY0);
: 219 0361 2
: 220 P 0362 2 $XABKEY_INIT(XAB=XABKEY0,
: 221 P 0363 2 KREF=0,
: 222 P 0364 2 POS0=0,
: 223 P 0365 2 SIZE=VMD_C_USERNAME, !Username is first 31 bytes
: 224 0366 2 NXT=0);
: 225 0367 2
: 226 0368 2 MAIL$T_SYSMAIFAB[FAB$L_XAB] = XABPRO; !Point FAB to the XAB chain
: 227 0369 2
: 228 0370 2 ! Open (or create) the file
: 229 0371 2
: 230 0372 2 MAIL$UP_PRIV(1^$BITPOSITION(PRV$V_SYSPRV),0);
: 231 0373 2 STATUS = $CREATE(FAB=MAIL$T_SYSMAIFAB,ERR=UTIL$REPORT_IO_ERROR);
: 232 0374 2 MAIL$DOWN_PRIV();
: 233 0375 2
: 234 0376 2 IF NOT .STATUS
: 235 0377 2 THEN RETURN .STATUS;
: 236 0378 2
: 237 P 0379 2 $RAB_INIT(RAB=MAIL$T_SYSMAIRAB,
: 238 P 0380 2 RAC=KEY,
: 239 P 0381 2 ROP=(RLK,NLK),
: 240 P 0382 2 KRF=0,
: 241 P 0383 2 KSZ=VMD_C_USERNAME,
: 242 P 0384 2 KBF=.MAIL$L_SYSMAIBUF,
: 243 P 0385 2 CTX=MAIL$ OPENOUT,
: 244 P 0386 2 FAB=MAIL$T_SYSMAIFAB,
: 245 P 0387 2 RBF=.MAIL$L_SYSMAIBUF,
: 246 P 0388 2 USZ=VMD_C_LENGTH+3*NAM$C_MAXRSS,
: 247 0389 2 UBF=.MAIL$L_SYSMAIBUF);
: 248 0390 2
: 249 0391 2 STATUS = $CONNECT(RAB=MAIL$T_SYSMAIRAB);
: 250 0392 2 IF NOT .STATUS
: 251 0393 3 AND NOT (IF .STATUS EQL RMS$ CRMP
: 252 0394 4 THEN STATUS = (MAIL$T_SYSMAIFAB[FAB$W_GBC] = 0;
: 253 0395 4 $CONNECT(RAB=MAIL$T_SYSMAIRAB))
: 254 0396 3 ELSE FALSE)
: 255 0397 3 THEN BEGIN
: 256 0398 3 UTIL$REPORT IO ERROR(MAIL$T_SYSMAIFAB);
: 257 0399 3 $CLOSE(FAB=MAIL$T_SYSMAIFAB);
: 258 0400 3 RETURN TRUE
: 259 0401 2 END;
: 260 0402 2 MAIL$T_SYSMAIRAB[RAB$L_CTX] = MAIL$ READERR;
: 261 0403 2 MAIL$T_SYSMAIFAB[FAB$L_XAB] = 0;
: 262 0404 2
: 263 0405 2 ! Read the user record if it exists
: 264 0406 2

```

```

265 0407 2 IF NOT .MAIL$GL_FLAGS[MAIF_V_NETJOB]
266 0408 3 THEN BEGIN
267 0409 3 CH$COPY(.MAIL$Q_USERNAME[DSC$W_LENGTH],.MAIL$Q_USERNAME[DSC$A_POINTER],
268 0410 3 ' ',VMD_C_USERNAME,MAIL$L_USR$SYSBUF[VMD_T_USERNAME]);
269 0411 3
270 0412 3 | Read the record. If it doesn't exist, then initialize the buffer
271 0413 3 | and write out the record for this user.
272 0414 3 |
273 0415 3 MAIL$T_SYSMAIRAB[RAB$L_KBF] = .MAIL$L_USR$SYSBUF;
274 0416 4 IF NOT $GET(RAB=MAIL$T_SYSMAIRAB)
275 0417 4 THEN BEGIN
276 0418 4 IF .MAIL$T_SYSMAIRAB[RAB$L_STS] NEQ RMS$RNF
277 0419 4 THEN RETURN UTIL$REPORT IO_ERROR(MAIL$T_SYSMAIRAB);
278 0420 4 CH$FILL(0,VMD_C_LENGTH+NAM$C_MAXRSS-$BYTEOFFSET(VMD_W_FLAGS),
279 0421 4 MAIL$L_USR$SYSBUF[VMD_W_FLAGS]);
280 0422 4 MAIL$UPDATE_SYSMAI_REC(.MAIL$L_USR$SYSBUF);
281 0423 4 END
282 0424 3 ELSE CH$MOVE(.MAIL$T_SYSMAIRAB[RAB$W_RSZ], !Copy/save this user
283 0425 3 .MAIL$T_SYSMAIRAB[RAB$L_RBF],
284 0426 3 .MAIL$L_USR$SYSBUF);
285 0427 2 END;
286 0428 2
287 0429 2 RETURN TRUE
288 0430 1 END;

```

				\$RMS_PTR=	MAIL\$T_SYSMAIRAB	
				\$RMS_PTR=	SYSMAIRAM	
				\$RMS_PTR=	MAIL\$T_SYSMAIRAB	
				.EXTRN	SYS\$CREATE, SYS\$CONNECT	
				.EXTRN	SYS\$CLOSE, SYS\$GET	
			OFFC 00000	.ENTRY	MAIL\$OPEN_SYSMAI, Save R2,R3,R4,R5,R6,R7,-	0293
					R8,R9,R10,R11	
				MOVAB	LIB\$GET_VM, R11	
				MOVAB	UTIL\$REPORT IO_ERROR, R10	
				MOVAB	VMSMAIL_DEST+4, R9	
				MOVAB	\$RMS_PTR, R8	
				MOVAB	\$RMS_PTR, R7	
				MOVAB	-168(SP), SP	
				TSTL	MAIL\$L_SYSMIBUF	0323
				BNEQ	1\$	
				PUSHAB	MAIL\$L_SYSMIBUF	0326
	04	AE	0341	MOVZWL	#833, 4(SP)	
			04	PUSHAB	4(SP)	
			6B	CALLS	#2, LIB\$GET_VM	
			17	BLBC	STATUS, 2\$	
				TSTL	MAIL\$L_USR\$SYSBUF	0327
				BNEQ	3\$	
				PUSHAB	MAIL\$L_USR\$SYSBUF	0330
	04	AE	0341	MOVZWL	#833, 4(SP)	
			04	PUSHAB	4(SP)	
			6B	CALLS	#2, LIB\$GET_VM	
			0C	BLBS	STATUS, 3\$	
				PUSHL	STATUS	
			0000000G 00	CALLS	#1, LIB\$SIGNAL	



			50	A7	9F	00178	PUSHAB	MAIL\$T_SYSMAIRAB	0391	
	00000000G	00		01	FB	0017B	CALLS	#1, SYS\$CONNECT		
		56		50	DO	00182	MOVL	R0, STATUS		
		2C		56	E8	00185	BLBS	STATUS, 6\$	0392	
	0001C14C	8F		56	D1	00188	CMPL	STATUS, #115020	0393	
				13	12	0018F	BNEQ	5\$		
			48	A7	B4	00191	CLRW	MAIL\$T_SYSMAIFAB+72	0394	
			50	A7	9F	00194	PUSHAB	MAIL\$T_SYSMAIRAB	0395	
	00000000G	00		01	FB	00197	CALLS	#1, SYS\$CONNECT		
		56		50	DO	0019E	MOVL	R0, STATUS		
		10		56	E8	001A1	BLBS	STATUS, 6\$		
				57	DD	001A4	5\$: PUSHL	R7	0398	
		6A		01	FB	001A6	CALLS	#1, UTIL\$REPORT_IO_ERROR		
				57	DD	001A9	PUSHL	R7	0399	
	00000000G	00		01	FB	001AB	CALLS	#1, SYS\$CLOSE		
				6E	11	001B2	BRB	9\$	0400	
	68	A7	007E10B0	8F	DO	001B4	6\$: MOVL	#8261808, MAIL\$T_SYSMAIRAB+24	0402	
			24	A7	D4	001BC	CLRL	MAIL\$T_SYSMAIFAB+36	0403	
	5B	00000000G	00	01	E0	001BF	BRB	#1, MAIL\$GL_FLAGS, 9\$	0407	
			50	00000000G	00	DO	001C7	MOVL	MAIL\$Q_USERNAME+4, R0	0409
			56	FC	A7	DO	001CE	MOVL	MAIL\$Q_USR\$SYSBUF, R6	0410
1F	20	60	00000000G	00	2C	001D2	MOVCS	MAIL\$Q_USERNAME, (R0), #32, #31, (R6)		
				66		001DB				
	0080	C7		56	DO	001DC	MOVL	R6, MAIL\$T_SYSMAIRAB+48	0415	
			50	A7	9F	001E1	PUSHAB	MAIL\$T_SYSMAIRAB	0416	
	00000000G	00		01	FB	001E4	CALLS	#1, SYS\$GET		
		27		50	E8	001EB	BLBS	R0, 8\$		
	000182B2	8F	58	A7	D1	001EE	CMPL	MAIL\$T_SYSMAIRAB+8, #98994	0418	
				07	13	001F6	BEQL	7\$		
			50	A7	9F	001F8	PUSHAB	MAIL\$T_SYSMAIRAB	0419	
		6A		01	FB	001FB	CALLS	#1, UTIL\$REPORT_IO_ERROR		
				04	001FE		RET			
		56	FC	A7	DO	001FF	7\$: MOVL	MAIL\$Q_USR\$SYSBUF, R6	0421	
0124	8F	00		00	2C	00203	MOVCS	#0, (SP), #0, #292, 31(R6)		
				1F	A6	0020A				
				56	DD	0020C	PUSHL	R6	0422	
	FD6C	CF		01	FB	0020E	CALLS	#1, MAIL\$UPDATE_SYSMAI_REC		
				0D	11	00213	BRB	9\$	0416	
		51	78	A7	DO	00215	8\$: MOVL	MAIL\$T_SYSMAIRAB+40, R1	0425	
		50	FC	A7	DO	00219	MOVL	MAIL\$Q_USR\$SYSBUF, R0	0426	
	60	61	72	A7	28	0021D	MOVCS	MAIL\$T_SYSMAIRAB+34, (R1), (R0)		
		50		01	DO	00222	9\$: MOVL	#1, R0	0429	
				04	00225		RET		0430	

; Routine Size: 550 bytes, Routine Base: \$CODE\$ + 00A9

```

290 0431 1 GLOBAL ROUTINE MAIL$CHECK_FORW(NAME_DESC,DATA_BUFFER) =
291 0432 2 BEGIN
292 0433 2 |**
293 0434 2 | FUNCTIONAL DESCRIPTION:
294 0435 2 |
295 0436 2 |     This routine checks if the named user has forwarding enabled
296 0437 2 |     in SYSS$SYSTEM:VMSMAIL.DAT
297 0438 2 |
298 0439 2 | INPUTS:
299 0440 2 |
300 0441 2 |     NAME_DESC = Address of string descriptor for user name
301 0442 2 |     DATA_BUFFER = Address of a buffer to read record in to (If not given,
302 0443 2 |                   MAIL$L_SYSMAIBUF will be used.
303 0444 2 |
304 0445 2 | OUTPUTS:
305 0446 2 |
306 0447 2 |     If found, the buffer pointed to by DATA_BUFFER contains
307 0448 2 |     the record.
308 0449 2 |
309 0450 2 | ROUTINE VALUE:
310 0451 2 |
311 0452 2 |     True if name is forwarded
312 0453 2 |     False if not
313 0454 2 | --
314 0455 2 BUILTIN
315 0456 2 NULLPARAMETER;
316 0457 2
317 0458 2 MAP
318 0459 2     NAME_DESC : REF $BLOCK,
319 0460 2     DATA_BUFFER : REF $BLOCK;
320 0461 2
321 0462 2 LOCAL
322 0463 2     BUFPTR : REF $BLOCK,
323 0464 2     STATUS;
324 0465 2
325 0466 2 IF NOT NULLPARAMETER(2)
326 0467 2     THEN BUFPTR = .DATA_BUFFER
327 0468 2     ELSE BUFPTR = .MAIL$L_SYSMAIBUF;
328 0469 2 CH$COPY(.NAME_DESC[DSC$W_LENGTH],.NAME_DESC[DSC$A_POINTER],
329 0470 2         ' ',VMD [ USERNAME, BUFPTR]);
330 0471 2 MAIL$T_SYSMAIRAB[RAB$L_OBF] = .BUFPTR;
331 0472 2 MAIL$T_SYSMAIRAB[RAB$L_KBF] = BUFPTR[VMD_T_USERNAME];
332 0473 2 MAIL$T_SYSMAIRAB[RAB$V_NLK] = TRUE;
333 0474 3 IF NOT $GET(RAB=MAIL$T_SYSMAIRAB)
334 0475 2     THEN RETURN 0;
335 0476 2
336 0477 2 RETURN (.BUFPTR[VMD_B_FWDLPG] NEQ 0);
337 0478 2
338 0479 1 END;

```

```

00FC 00000
57 00000000' 00 9E 00002
02          6C 91 00009

```

```

.ENTRY MAIL$CHECK_FORW, Save R2,R3,R4,R5,R6,R7 : 0431
MOVAB MAIL$L_SYSMAIBUF, R7 :
CMPB (AP), #2 : 0466

```



```

340 0480 1 GLOBAL ROUTINE MAIL$GET_SYSMAI_REC(NAME_DESC,RECBUF) =
341 0481 1 ++
342 0482 1 FUNCTIONAL DESCRIPTION:
343 0483 1
344 0484 1 Read the record from VMSMAIL.DAT for the named user
345 0485 1
346 0486 1 INPUTS:
347 0487 1
348 0488 1 NAME_DESC = Address of descriptor for username
349 0489 1 RECBUF = Address of buffer to return record in
350 0490 1
351 0491 1 --
352 0492 2 BEGIN
353 0493 2 LOCAL
354 0494 2 STATUS;
355 0495 2 MAP
356 0496 2 NAME_DESC : REF $BBLOCK,
357 0497 2 RECBUF : REF $BBLOCK;
358 0498 2
359 0499 2 CH$COPY(.NAME_DESC[DSC$W_LENGTH],.NAME_DESC[DSC$A_POINTER],
360 0500 2 ' ' ,VMD [ USERNAME ,RECBUF]);
361 0501 2 MAIL$T_SYSMAIRAB[RAB$L_OBF] = .RECBUF;
362 0502 2 MAIL$T_SYSMAIRAB[RAB$L_KBF] = RECBUF[VMD_T_USERNAME];
363 0503 2 MAIL$T_SYSMAIRAB[RAB$V_NLK] = TRUE;
364 0504 2 STATUS = $GET(RAB=MAIL$T_SYSMAIRAB);
365 0505 2
366 0506 2 RETURN .STATUS
367 0507 1 END;

```

				007C 00000	.ENTRY MAIL\$GET SYSMAI REC, Save R2,R3,R4,R5,R6	: 0480
		56 00000000'	00	9E 00002	MOVAB MAIL\$T_SYSMAIRAB+36, R6	: 0499
1F	20	04 B0	AC	DO 00009	MOVL NAME_DESC, R0	: 0500
			60	2C 0000D	MOVCS (R0), @4(R0), #32, #31, @RECBUF	: 0501
			08	BC 00013		: 0502
		66	08	AC DO 00015	MOVL RECBUF, MAIL\$T_SYSMAIRAB+36	: 0503
		0C A6	08	AC DO 00019	MOVL RECBUF, MAIL\$T_SYSMAIRAB+48	: 0504
		E2 A6	10	88 0001E	BISB2 #16, MAIL\$T_SYSMAIRAB+6	: 0505
			DC	A6 9F 00022	PUSHAB MAIL\$T_SYSMAIRAB	: 0506
		00000000G 00	01	FB 00025	CALLS #1, SYS\$GET	: 0507
			04	0002C	RET	: 0507

; Routine Size: 45 bytes, Routine Base: \$CODE\$ + 031D



```

369 0508 1 GLOBAL ROUTINE MAIL$GETLK_SYMAI_REC (NAME_DESC, RECDISC) =
370 0509 1 ++
371 0510 1 FUNCTIONAL DESCRIPTION:
372 0511 1
373 0512 1     Read and lock the designated record in VMSMAIL
374 0513 1
375 0514 1
376 0515 1 INPUTS:
377 0516 1
378 0517 1     NAME_DESC = Address of descriptor for username
379 0518 1     RECDISC = Address of descriptor to return record descriptor in
380 0519 1
381 0520 1 --
382 0521 2 BEGIN
383 0522 2 MAP
384 0523 2     NAME_DESC : REF $BLOCK,
385 0524 2     RECDISC : REF VECTOR[ ,LONG];
386 0525 2
387 0526 2 LOCAL
388 0527 2     COUNTER,
389 0528 2     STATUS;
390 0529 2
391 0530 2 LITERAL
392 0531 2     RETRY_RLK = 10,                    ! number of retries for a locked record
393 0532 2     SLEEP_RLK = 75;                 ! ms to sleep before retrying
394 0533 2
395 0534 2 BIND
396 0535 2     WAKEDELTA = UPLIT LONG(-10*1000*SLEEP_RLK,-1);
397 0536 2
398 0537 2 CH$COPY(.NAME_DESC[DSCSW_LENGTH],.NAME_DESC[DSCSA_POINTER],
399 0538 2     ,.VMD [.USERNAME,.MAIL$SYMAIBUF]);
400 0539 2 MAIL$SYMAIRAB[RAB$R_OBF] = .MAIL$SYMAIBUF;
401 0540 2 MAIL$SYMAIRAB[RAB$R_KBF] = MAIL$SYMAIBUF[VMD_T_USERNAME];
402 0541 2 MAIL$SYMAIRAB[RAB$V_NLK] = FALSE;
403 0542 2 COUNTER = RETRY_RLK;
404 0543 3 WHILE ((STATUS = $GET(RAB=MAIL$SYMAIRAB)) EQL RMSS_RLK)
405 0544 3     AND ((COUNTER = .COUNTER - 1) GEQ 0)
406 0545 2     DO IF $SCHDWK(DAYTIM = WAKEDELTA) THEN $HIBER;
407 0546 2
408 0547 2 IF NOT .STATUS
409 0548 2     THEN RETURN .STATUS;
410 0549 2
411 0550 2 RECDISC[0] = .MAIL$SYMAIRAB[RAB$R_RSZ];
412 0551 2 RECDISC[1] = .MAIL$SYMAIRAB[RAB$R_RBF];
413 0552 2
414 0553 2 RETURN .STATUS
415 0554 1 END;
```

```

0034A .BLKB 2
FFFFFFFF FFF48E50 0034C P.AAE: .LONG -750000, -1 ;
```

```

WAKEDELTA= P.AAE
.EXTRN SYSSCHDWK, SYSSHIBER
```

```
00FC 0000 .ENTRY MAIL$GETLK_SYMAI_REC, Save R2,R3,R4,R5,R6,-; 0508
```

			57	00000000'	00	9E	00002		MOVAB	R7		
			50	04	AC	D0	00009		MOVAB	MAILSL_SYSMAIBUF, R7		0537
			56		67	D0	0000D		MOVL	NAME_DESC, R0		0538
1F	20	04	B0		60	2C	00010		MOVL	MAILSL_SYSMAIBUF, R6		
					66		00016		MOVCS	(R0), #4(R0), #32, #31, (R6)		
		7C	A7		56	D0	00017		MOVL	R6, MAILST_SYSMAIRAB+36		0539
		0088	C7		56	D0	0001B		MOVL	R6, MAILST_SYSMAIRAB+48		0540
		5E	A7		10	8A	00020		BICB2	#16, MAILST_SYSMAIRAB+6		0541
			53		0A	D0	00024		MOVL	#10, COUNTER		0542
				58	A7	9F	00027	1\$:	PUSHAB	MAILST_SYSMAIRAB		0543
		00000000G	00		01	FB	0002A		CALLS	#1, SYSSGET		
		000182AA	52		50	D0	00031		MOVL	R0, STATUS		
			8F		52	D1	00034		CMFL	STATUS, #98986		
					1E	12	0003B		BNEQ	2\$		
					53	D7	0003D		DECL	COUNTER		0544
					1A	19	0003F		BLSS	2\$		
				B2	7E	D4	00041		CLRL	-(SP)		0545
					AF	9F	00043		PUSHAB	WAKEDELTA		
					7E	7C	00046		CLRQ	-(SP)		
		00000000G	00		04	FB	00048		CALLS	#4, SYSSCHUNK		
			D5		50	E9	0004F		BLBC	R0, 1\$		
		00000000G	00		00	FB	00052		CALLS	#0, SYSSHIBER		
					CC	11	00059		BRB	1\$		
			0E		52	E9	0005B	2\$:	BLBC	STATUS, 3\$		0547
			50	08	AC	D0	0005E		MOVL	RECDESC, R0		0550
			60	7A	A7	3C	00062		MOVZWL	MAILST_SYSMAIRAB+34, (R0)		
		04	A0	0080	C7	D0	00066		MOVL	MAILST_SYSMAIRAB+40, 4(R0)		0551
			50		52	D0	0006C	3\$:	MOVL	STATUS, R0		0553
					04	04	0006F		RET			0554

; Routine Size: 112 bytes, Routine Base: \$CODE\$ + 0354

```

: 417 0555 1 GLOBAL ROUTINE MAIL$GET_DEVDIR(USERDESC,DEVDIRDESC) =
: 418 0556 1 |**
: 419 0557 1 | FUNCTIONAL DESCRIPTION:
: 420 0558 1 |
: 421 0559 1 |     This routine gets the device/directory descriptor for USERNAME
: 422 0560 1 |     and returns it in DEVDIRDESC via STR$COPY.
: 423 0561 1 |
: 424 0562 1 | --
: 425 0563 2 BEGIN
: 426 0564 2 MAP
: 427 0565 2     USERDESC : REF $BBLOCK,
: 428 0566 2     DEVDIRDESC : REF $BBLOCK;
: 429 0567 2
: 430 0568 2 LOCAL
: 431 0569 2     STATUS,
: 432 0570 2     DIRBUF : $BBLOCK[NAM$C_MAXRSS],
: 433 0571 2     PTR : REF VECTOR[BYTE],
: 434 0572 2     DESC : VECTOR[2,LONG],
: 435 0573 2     SYSMAIBUF : $BBLOCK[VMD_C_LENGTH+3*NAM$C_MAXRSS];
: 436 0574 2
: 437 0575 2 PTR = DIRBUF;
: 438 0576 3 IF NOT (STATUS = MAIL$GET_USER(.USERDESC))
: 439 0577 2 THEN RETURN .STATUS
: 440 0578 3 ELSE BEGIN
: 441 0579 3     PTR = CH$MOVE(. (MAIL$T_UAF_BUFF[UAF$T_DEFDEV])<CS_COUNT>,
: 442 0580 3         (MAIL$T_UAF_BUFF[UAF$T_DEFDEV])<CS_STRING>,
: 443 0581 3         DIRBUF);
: 444 0582 3     PTR = CH$MOVE(. (MAIL$T_UAF_BUFF[UAF$T_DEFDIR])<CS_COUNT>,
: 445 0583 3         (MAIL$T_UAF_BUFF[UAF$T_DEFDIR])<CS_STRING>,.PTR);
: 446 0584 2     END;
: 447 0585 2 IF MAIL$GET_SYSMAI_REC(.USERDESC,SYSMAIBUF)
: 448 0586 2 AND .SYSMAIBUF[VMD_B_DIRLNG] NEQ 0
: 449 0587 2 THEN PTR = CH$MOVE(.SYSMAIBUF[VMD_B_DIRLNG]-1,
: 450 0588 2     SYSMAIBUF[VMD_T_FWDNAM]+.SYSMAIBUF[VMD_B_FWDLNG]+
: 451 0589 2     .SYSMAIBUF[VMD_B_FNMLNG]+1,.PTR-T);
: 452 0590 2
: 453 0591 2 IF .((MAIL$T_UAF_BUFF[UAF$T_DEFDIR])<CS_STRING>)<0,8> EQL %C'[
: 454 0592 2     THEN PTR[-1] = %C']'
: 455 0593 2     ELSE PTR[-1] = %C'>';
: 456 0594 2 DESC[0] = .PTR - DIRBUF;
: 457 0595 2 DESC[1] = DIRBUF;
: 458 0596 2 STR$COPY_DX(.DEVDIRDESC,DESC);
: 459 0597 2
: 460 0598 2 RETURN TRUE
: 461 0599 1 END;

```

		00FC 0000	.ENTRY MAIL\$GET_DEVDIR, Save R2,R3,R4,R5,R6,R7	: 0555
57	00000000G	00 9E 00002	MOVAB MAIL\$T_UAF_BUFF, R7	:
5E	FBB4	CE 9E 00009	MOVAB -1100(SP), -SP	:
53	FF00	CD 9E 0000E	MOVAB DIRBUF, PTR	: 0575
	04	AC DD 00013	PUSHL USERDESC	: 0576
00000000G	00	01 FB 00016	CALLS #1, MAIL\$GET_USER	:
	7C	50 E9 0001D	BLBC STATUS, 4\$	:

			56		67	90	J0020		MOVL	MAILST UAF BUFF, R6	:	0579
			50	74	A6	9A	00023		MOVZBL	116(R6), R0	:	
FF00	CD	75	A6		50	28	00027		MOV3	R0, 117(R6), DIRBUF	:	
			50	0094	C6	9A	0002E		MOVZBL	148(R6), R0	:	0582
	63	0095	C6		50	28	00033		MOV3	R0, 149(R6), (PTR)	:	0583
					5F	DD	00039		PUSHL	SP	:	0585
				04	AC	DD	0003B		PUSHL	USERDESC	:	
		FF16	CF		02	FB	0003E		CALLS	#2, MAIL\$GET_SYSMAI_REC	:	
			1D		50	E9	00043		BLBC	R0, 1\$	:	
				41	AE	95	00046		TSTB	SYSMAIBUF+65	:	0586
					18	13	00049		BEQL	1\$	:	
			51	41	AE	9A	0004B		MOVZBL	SYSMAIBUF+65, R1	:	0587
					51	D7	0004F		DECL	R1	:	
			50	43	AE	9A	00051		MOVZBL	SYSMAIBUF+67, R0	:	0589
			52	42	AE	9A	00055		MOVZBL	SYSMAIBUF+66, R2	:	
			50		52	C0	00059		ADDL2	R2, R0	:	
FF	A3	45	AE40		51	28	0005C		MOV3	R1, SYSMAIBUF+69[R0], -1(PTR)	:	
			50		67	D0	00063	1\$:	MOVL	MAILST UAF BUFF, R0	:	0591
		5B	8F	0095	C0	91	00066		CMPB	149(R0), #91	:	
					07	12	0006C		BNEQ	2\$	:	
		FF	A3	5D	8F	90	0006E		MOVB	#93, -1(PTR)	:	0592
					04	11	00073		BRB	3\$	:	
		FF	A3		3E	90	00075	2\$:	MOVB	#62, -1(PTR)	:	0593
			50	FF00	CD	9E	00079	3\$:	MOVAB	DIRBUF, R0	:	0594
FEF8	CD		53		50	C3	0007E		SUBL3	R0, PTR, DESC	:	
		FEFC	CD	FF00	CD	9E	00084		MOVAB	DIRBUF, DESC+4	:	0595
				FEF8	CD	9F	0008B		PUSHAB	DESC	:	0596
				08	AC	DD	0008F		PUSHL	DEVDIRDESC	:	
		00000000G	00		02	FB	00092		CALLS	#2, STR\$COPY_DX	:	
			50		01	D0	00099		MOVL	#1, R0	:	0598
					04	0009C	4\$:		RET		:	0599

; Routine Size: 157 bytes, Routine Base: \$CODE\$ + 03C4

```

: 463      0600 1 GLOBAL ROUTINE MAIL$GET_NEWMAIL(USERDESC,COUNT) =
: 464      0601 1 |++
: 465      0602 1 | FUNCTIONAL DESCRIPTION:
: 466      0603 1 |
: 467      0604 1 |         Return the new mail count from VMSMAIL
: 468      0605 1 |
: 469      0606 1 | --
: 470      0607 2 BEGIN
: 471      0608 2
: 472      0609 2 LOCAL
: 473      0610 2     PTR : REF $BBLOCK,
: 474      0611 2     SYSMAIBUF : $BBLOCK[VMD_C_LENGTH+3*NAM$C_MAXRSS];
: 475      0612 2
: 476      0613 2     .COUNT = 0;
: 477      P 0614 2 IF_ERR(MAIL$GET_SYSMAI_REC(.USERDESC,SYSMAIBUF);,
: 478      0615 2     RETURN TRUE);
: 479      0616 2
: 480      0617 2     .COUNT = .SYSMAIBUF[VMD_W_MAIL];
: 481      0618 2
: 482      0619 2 RETURN TRUE
: 483      0620 1 END;

```

```

          SE      FCBC      0000 0000      .ENTRY MAIL$GET_NEWMAIL, Save nothing      : 0600
          08      BC      D4 00002      MOVAB  -836(SP), SP      :
          04      SE      DD 00007      CLRL  @COUNT      : 0613
          FEAB    CF      AC      DD 0000A      PUSHL SP      : 0615
          08      05      02      FB 0000C      PUSHL USERDESC
          BC      21      50      E9 0000F      CALLS #2, MAIL$GET_SYSMAI_REC
          50      AE      3C 00014      BLBC  STATUS, 1$
          01      D0 00017      MOVZWL SYSMAIBUF+33, @COUNT
          04      0001C 1$:      MOVL  #1, R0
          0001F      RET      : 0617
          : 0619
          : 0620

```

; Routine Size: 32 bytes, Routine Base: \$CODE\$ + 0461

```

485 0621 1 GLOBAL ROUTINE MAIL$INCR_NEWMAIL(USERDESC,COUNT) =
486 0622 1 |++
487 0623 1 | FUNCTIONAL DESCRIPTION:
488 0624 1 |
489 0625 1 |     Increment the new mail count of the specified user
490 0626 1 |
491 0627 1 | Inputs:
492 0628 1 |
493 0629 1 |     Userdesc = address of descriptor of user to update
494 0630 1 |     COUNT = (optional) address of value to set the count to
495 0631 1 |
496 0632 1 | --
497 0633 2 BEGIN
498 0634 2 BUILTIN
499 0635 2     ACTUALCOUNT;
500 0636 2
501 0637 2 MAP
502 0638 2     USERDESC : REF $BLOCK;
503 0639 2
504 0640 2 LOCAL
505 0641 2     PTR : REF $BLOCK,
506 0642 2     SYSMAIBUF : $BLOCK[VMD_C_LENGTH],
507 0643 2     DESC : VECTOR[2, LONG];
508 0644 2
509 0645 3 RETURN (IF MAIL$GETLK_SYSMAI_REC(.USERDESC, DESC)
510 0646 4 THEN BEGIN
511 0647 4     PTR = .DESC[1];
512 0648 4     IF ACTUALCOUNT() LSSU 2
513 0649 4         THEN PTR[VMD_W_MAIL] = .PTR[VMD_W_MAIL] + 1
514 0650 4         ELSE PTR[VMD_W_MAIL] = .COUNT;
515 0651 4     MAIL$UPDATE_SYSMAI_REC(.DESC[1])
516 0652 4     END
517 0653 4 ELSE BEGIN
518 0654 4     CH$FILL(0, VMD_C_LENGTH, SYSMAIBUF);
519 0655 4     CH$COPY(.USERDESC[DSC$W_LENGTH], .USERDESC[DSC$A_POINTER],
520 0656 4         ' ', VMD_C_USERNAME, SYSMAIBUF);
521 0657 5     SYSMAIBUF[VMD_W_MAIL] = (IF ACTUALCOUNT() LSSU 2
522 0658 5         THEN 1
523 0659 4         ELSE .COUNT);
524 0660 4     MAIL$UPDATE_SYSMAI_REC(SYSMAIBUF)
525 0661 3     END)
526 0662 1 END;

```

			007C 0000	.ENTRY	MAIL\$INCR_NEWMAIL, Save R2,R3,R4,R5,R6	: 0621
	5E	B4	AE 9E 0002	MOVAB	-76(SP), SP	
			5E DD 0006	PUSHL	SP	: 0645
	56	04	AC D0 0008	MOVL	USERDESC, R6	
			56 DD 000C	PUSHL	R6	
FECO	CF		02 FB 000E	CALLS	#2, MAIL\$GETLK_SYSMAI_REC	
	18		50 E9 0013	BLBC	R0, 3\$	
	50	04	AE D0 0016	MOVL	DESC+4, PTR	: 0647
	02		6C 91 001A	CMPB	(AP), #2	: 0648
			05 1E 001D	BGEQU	1\$	

MAIL\$SYSMAI  
V04-000

OPEN/READ/WRITE SYS\$SYSTEM:VMSMAIL.DAT

G 16  
16-Sep-1984 01:23:17  
14-Sep-1984 12:42:35

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[MAIL.SRC]SYSMAI.B32;1

Page 21  
(10)

				21	A0	B6	0001F		INCW	33(PTR)	:	0649
				05	11	00022			BRB	2\$	:	
		21	A0	08	AC	B0	00024	1\$:	MOVW	COUNT, 33(PTR)	:	0650
				04	AE	DD	00029	2\$:	PUSHL	DESC+4	:	0651
					26	11	0002C		BRB	6\$	:	
0044	8F		00		00	2C	0002E	3\$:	MOVCS	#0, (SP), #0, #68, SYSMAIBUF	:	0654
					08	AE	00035				:	
	1F		20	04	B6	2C	00037		MOVCS	(R6), @4(R6), #32, #31, SYSMAIBUF	:	0655
					08	AE	0003D				:	
					02	6C	91	0003F	CMPB	(AP), #2	:	0657
					05	1E	00042		BGEQU	4\$	:	
					50	01	D0	00044	MOVL	#1, R0	:	
					04	11	00047		BRB	5\$	:	
					50	08	AC	D0	00049	4\$:		0659
		29	AE		50	B0	0004D	5\$:	MOVL	COUNT, R0	:	0657
					08	AE	9F	00051	MOVW	R0, SYSMAIBUF+33	:	0657
									PUSHAB	SYSMAIBUF	:	0660
		FB4E	CF		01	FB	00054	6\$:	CALLS	#1, MAIL\$UPDATE_SYSMAI_REC	:	
					04	04	00059		RET		:	0662

; Routine Size: 90 bytes, Routine Base: \$CODE\$ + 0481

```

: 528 0663 1 GLOBAL ROUTINE MAIL$DECR_NEWMAIL(USERDESC) =
: 529 0664 1 |++
: 530 0665 1 | FUNCTIONAL DESCRIPTION:
: 531 0666 1 |
: 532 0667 1 |     Decrement the new mail count
: 533 0668 1 |
: 534 0669 1 | |--
: 535 0670 2 BEGIN
: 536 0671 2
: 537 0672 2 MAP
: 538 0673 2     USERDESC : REF $BBLOCK;
: 539 0674 2
: 540 0675 2 LOCAL
: 541 0676 2     PTR : REF $BBLOCK,
: 542 0677 2     RECDISC : VECTOR[2, LONG];
: 543 0678 2
: 544 0679 2 IF MAIL$GETLK_SYSMAI_REC(.USERDESC, RECDISC)
: 545 0680 3 THEN BEGIN
: 546 0681 3     PTR = .RECDISC[1];
: 547 0682 3     IF .PTR[VMD_W_MAIL] GTRU 0
: 548 0683 3         THEN PTR[VMD_W_MAIL] = .PTR[VMD_W_MAIL] - 1;
: 549 0684 3     MAIL$UPDATE_SYSMAI_REC(.RECDISC[1])
: 550 0685 2     END;
: 551 0686 2
: 552 0687 2 RETURN TRUE
: 553 0688 1 END;

```

			0000 0000	.ENTRY	MAIL\$DECR_NEWMAIL, Save nothing	: 0663
	SE		08 C2 00002	SUBL2	#8, SP	:
			5E DD 00005	PUSHL	SP	: 0679
		04	AC DD 00007	PUSHL	USERDESC	:
FE6A	CF		02 FB 0000A	CALLS	#2, MAIL\$GETLK_SYSMAI_REC	:
	14		50 E9 0000F	BLBC	R0, 2\$	:
	50	04	AE D0 00012	MOVL	RECDISC+4, PTR	: 0681
		21	A0 B5 00016	TSTW	33(PTR)	: 0682
			03 13 00019	BEQL	1\$	:
		21	A0 B7 0001B	DECW	33(PTR)	: 0683
		04	AE DD 0001E 1\$:	PUSHL	RECDISC+4	: 0684
FB27	CF		01 FB 00021	CALLS	#1, MAIL\$UPDATE_SYSMAI_REC	:
	50		01 D0 00026 2\$:	MOVL	#1, R0	: 0687
			04 00029	RET		: 0688

: Routine Size: 42 bytes, Routine Base: \$CODE\$ + 04DB



MAIL\$SYSMAI  
V04-000

OPEN/READ/WRITE SYSS\$SYSTEM:VMSMAIL.DAT

I 16  
16-Sep-1984 01:23:17  
14-Sep-1984 12:42:35

VAX-11 Bliss-32 V4.0-742 Page 23  
DISK\$VMSMASTER:[MAIL.SRC]SYSMAI.B32;1 (12)

: 555 0689 1 END  
: 556 0690 0 ELUDOM

.EXTRN LIB\$SIGNAL

PSECT SUMMARY

Name	Bytes	Attributes
\$GLOBALS	156 NOVEC, WRT, RD	,NOEXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$CODE\$	1285 NOVEC,NOWRT, RD	, EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
\$OWNS	352 NOVEC, WRT, RD	,NOEXE,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	180 0	1000	00:01.4
_\$255\$DUA28:[MAIL.OBJ]MAILDEF.L32;1	457	10 2	26	00:00.4

COMMAND QUALIFIERS

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

: Size: 1235 code + 558 data bytes  
: Run Time: 00:19.9  
: Elapsed Time: 01:22.9  
: Lines/CPU Min: 2081  
: Lexemes/CPU-Min: 47453  
: Memory Used: 236 pages  
: Compilation Complete



0231 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

