



```

PPPPPPPP      AAAAAA      SSSSSSSS  RRRRRRRR  EEEEEEEEEEE  WW      WW  RRRRRRRR  IIIIII  TTTTTTTTTT
PPPPPPPP      AAAAAA      SSSSSSSS  RRRRRRRR  EEEEEEEEEEE  WW      WW  RRRRRRRR  IIIIII  TTTTTTTTTT
PP      PP  AA      AA  SS      RR      RR  EEEEEEEEEEE  WW      WW  RRRRRRRR  II      TT
PP      PP  AA      AA  SS      RR      RR  EEEEEEEEEEE  WW      WW  RRRRRRRR  II      TT
PP      PP  AA      AA  SS      RR      RR  EEEEEEEEEEE  WW      WW  RRRRRRRR  II      TT
PP      PP  AA      AA  SS      RR      RR  EEEEEEEEEEE  WW      WW  RRRRRRRR  II      TT
PPPPPPPP      AA      AA  SSSSSS  RRRRRRRR  EEEEEEEEE  WW      WW  RRRRRRRR  II      TT
PPPPPPPP      AA      AA  SSSSSS  RRRRRRRR  EEEEEEEEE  WW      WW  RRRRRRRR  II      TT
PP      AAAAAAAAAA      SS      RR  RR  EEEEEEEEE  WW  WW  WW  RR  RR  II      TT
PP      AAAAAAAAAA      SS      RR  RR  EEEEEEEEE  WW  WW  WW  RR  RR  II      TT
PP      AA      AA  SS      RR      RR  EEEEEEEEE  WWWW  WWWW  RR  RR  II      TT
PP      AA      AA  SS      RR      RR  EEEEEEEEE  WWWW  WWWW  RR  RR  II      TT
PP      AA      AA  SSSSSSSS  RR      RR  EEEEEEEEEEE  WW      WW  RR      RR  IIIIII  TT
PP      AA      AA  SSSSSSSS  RR      RR  EEEEEEEEEEE  WW      WW  RR      RR  IIIIII  TT

```

```

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
LLLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLLL  IIIIII  SSSSSSSS

```

```

1 0001 0 MODULE PASS$REWRITE2 ( %TITLE 'REWRITE procedure'
2 0002 0 IDENT = '1-004' ! File: PASREWRI.B32 Edit: SBL1004
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
10 0010 1 * ALL RIGHTS RESERVED. *
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
17 0017 1 * TRANSFERRED. *
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
21 0021 1 * CORPORATION. *
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: Pascal Language Support
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 This module contains PASS$REWRITE2, which implements the
36 0036 1 VAX-11 Pascal REWRITE procedure.
37 0037 1
38 0038 1 ENVIRONMENT: User mode - AST reentrant
39 0039 1
40 0040 1 AUTHOR: Steven B. Lionel, CREATION DATE: 1-April-1981
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. SBL 1-April-1981
45 0045 1 1-002 - Set FAB$V TEF if $TRUNCATE successful on sequential files. SBL 8-July-1982
46 0046 1 1-003 - Don't $REWIND, etc., unless file is a disk or tape. SBL 29-July-1982
47 0047 1 1-004 - Clear RAB$V_NLK. SBL 8-Apr-1983
48 0048 1 --
49 0049 1

```

```
51 0050 1 %SBTTL 'Declarations'  
52 0051 1  
53 0052 1 : PROLOGUE DEFINITIONS:  
54 0053 1 :  
55 0054 1  
56 0055 1 REQUIRE 'RTLIN:PASPROLOG'; . Externals, linkages, PSECTs, structures  
57 0119 1  
58 0120 1 :  
59 0121 1 : TABLE OF CONTENTS:  
60 0122 1 :  
61 0123 1  
62 0124 1 FORWARD ROUTINE  
63 0125 1 PASS$REWRITE2: NOVALUE; ! Do a REWRITE  
64 0126 1  
65 0127 1 :  
66 0128 1 : MACROS:  
67 0129 1 :  
68 0130 1 : NONE  
69 0131 1 :  
70 0132 1 : EQUATED SYMBOLS:  
71 0133 1 :  
72 0134 1 : NONE  
73 0135 1 :  
74 0136 1 : FIELDS:  
75 0137 1 :  
76 0138 1 : NONE  
77 0139 1 :  
78 0140 1 : OWN STORAGE:  
79 0141 1 :  
80 0142 1 : NONE
```

```

82 0143 1 %SBTTL 'PASS$REWRITE2 - REWRITE procedure'
83 0144 1 GLOBAL ROUTINE PASS$REWRITE2 (           ! Do a REWRITE
84 0145 1     PFV: REF $PASS$PFV_FILE_VARIABLE,     ! File variable
85 0146 1     ERROR                                 ! Error unwind address
86 0147 1     ): NOVALUE =
87 0148 1
88 0149 1 !++
89 0150 1 ! FUNCTIONAL DESCRIPTION:
90 0151 1
91 0152 1     PASS$REWRITE2 implements the VAX-11 Pascal REWRITE procedure. It
92 0153 1     opens the file if not already open, deletes all data from the file,
93 0154 1     and enters Generation mode.
94 0155 1
95 0156 1 ! CALLING SEQUENCE:
96 0157 1
97 0158 1     CALL PASS$FEWRITE2 (PFV.mr.r [, ERROR.j.r])
98 0159 1
99 0160 1 ! FORMAL PARAMETERS:
100 0161 1
101 0162 1     PFV           - The Pascal File Variable (PFV) passed by reference.
102 0163 1     The structure of the PFV is defined in PAS$PFV.REQ.
103 0164 1
104 0165 1     ERROR       - Optional. If specified, the address to unwind to in
105 0166 1     case of an error.
106 0167 1
107 0168 1 ! IMPLICIT INPUTS:
108 0169 1
109 0170 1     NONE
110 0171 1
111 0172 1 ! IMPLICIT OUTPUTS:
112 0173 1
113 0174 1     NONE
114 0175 1
115 0176 1 ! ROUTINE VALUE:
116 0177 1
117 0178 1     NONE
118 0179 1
119 0180 1 ! SIDE EFFECTS:
120 0181 1
121 0182 1     Throws away partial line, if any.
122 0183 1     Deletes all data from the file.
123 0184 1     Positions to the beginning of the file.
124 0185 1     Enters Generation mode.
125 0186 1
126 0187 1 ! SIGNALLED ERRORS:
127 0188 1
128 0189 1     REWNOTALL - REWRITE not allowed without exclusive access
129 0190 1     ERRDURREW - error during REWRITE
130 0191 1     GENNOTALL - Generation mode not allowed
131 0192 1
132 0193 1 --
133 0194 1
134 0195 2     BEGIN
135 0196 2
136 0197 2     LOCAL
137 0198 2     FCB: REF $PASS$FCB_CONTROL_BLOCK,           ! File control block
138 0199 2     PFV_ADDR: VOLATILE,                         ! Enable argument

```

```

139 0200 2      UNWIND_ACT: VOLATILE,          ! Enable argument
140 0201 2      ERROR_ADDR: VOLATILE;      ! Enable argument
141 0202 2
142 0203 2      BIND
143 0204 2      RAB = FCB: REF BLOCK [, BYTE],      ! RAB is also FCB address
144 0205 2      FAB = FCB: REF $PASS$FAB_FCB_STRUCT, ! FAB is based on FCB
145 0206 2      NAM = FCB: REF $PASS$NAM_FCB_STRUCT; ! NAM is based on FCB
146 0207 2
147 0208 2      BUILTIN
148 0209 2      ACTUALCOUNT,
149 0210 2      TESTBITSS;
150 0211 2
151 0212 2      ENABLE
152 0213 2      PASS$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR); ! Enable error handler
153 0214 2
154 0215 2      IF ACTUALCOUNT () GEQU 2
155 0216 2      THEN
156 0217 2      ERROR_ADDR = .ERROR;          ! Set unwind address
157 0218 2
158 0219 2      !+
159 0220 2      ! Set enable argument for PFV address.
160 0221 2      !-
161 0222 2
162 0223 2      PFV_ADDR = PFV [PFV$R_PFV];
163 0224 2
164 0225 2      !+
165 0226 2      ! Validate and lock PFV.
166 0227 2      !-
167 0228 2
168 0229 2      PASS$VALIDATE_PFV (PFV [PFV$R_PFV]; FCB);
169 0230 2
170 0231 2      !+
171 0232 2      ! Set unwind action to unlock file.
172 0233 2      !-
173 0234 2
174 0235 2      UNWIND_ACT = PASS$K_UNWIND_UNLOCK;
175 0236 2
176 0237 2      !+
177 0238 2      ! Open file if it should be implicitly opened.
178 0239 2      !-
179 0240 2
180 0241 2      IF NOT .PFV [PFV$V_VALID]
181 0242 2      THEN
182 0243 2      PASS$OPEN_IMPLICIT (PFV [PFV$R_PFV], FCB [FCB$R_FCB]; FCB);
183 0244 2
184 0245 2      !+
185 0246 2      ! See if the file is open. If not, open it.
186 0247 2      !-
187 0248 2
188 0249 2      IF NOT .PFV [PFV$V_OPEN]          ! Not open
189 0250 2      THEN
190 0251 2      PASS$OPEN (PFV [PFV$R_PFV], PASS$K_HISTORY_NEW; FCB);
191 0252 2
192 0253 2      !+
193 0254 2      ! If the file has the NOWRITE attribute, signal an error.
194 0255 2      !-
195 0256 2

```

```

196 0257 2 IF .FCB [FCB$V_NOWRITE]
197 0258 2 THEN
198 0259 2 $PASSIO_ERROR (PASS_GENNOTALL,0); ! Generation mode not allowed
199 0260 2
200 0261 2 !+
201 0262 2 ! If a textfile, and if in Generation mode, and if the buffer
202 0263 2 ! is not empty, do a Writeln.
203 0264 2 !-
204 0265 2
205 0266 2 IF .FCB [FCB$V_TEXT] AND .FCB [FCB$V_GENERATION] AND
206 0267 2 (.FCB [FCB$A_RECORD_CUR] GTRA .FCB [FCB$A_RECORD_BEG])
207 0268 2 THEN
208 0269 2 PASS$Writeln (PFV [PFV$R_PFV], FCB [FCB$R_FCB]);
209 0270 2
210 0271 2 !+
211 0272 2 ! Cancel any lazy lookahead in progress.
212 0273 2 !-
213 0274 2
214 0275 2 FCB [FCB$V_LAZY] = 0;
215 0276 2
216 0277 2 !+
217 0278 2 ! Clear RAB$V_NLK which may have been set by PASS$GET_UNLOCK.
218 0279 2 !-
219 0280 2
220 0281 2 RAB [RAB$V_NLK] = 0;
221 0282 2
222 0283 2 !+
223 0284 2 ! Is the device a file oriented device and not process-permanent?
224 0285 2 ! If so, proceed with rewind.
225 0286 2 !-
226 0287 2
227 0288 2 IF .FCB [FCB$V_FOD] AND NOT .NAM [NAM$V_PPF]
228 0289 2 THEN
229 0290 2 BEGIN
230 0291 2
231 0292 2 !+
232 0293 2 ! If this is a sequential organization file, we rewind and truncate.
233 0294 2 ! For relative and indexed organizations, we must rewind and delete
234 0295 2 ! every record.
235 0296 2 !-
236 0297 2
237 0298 2 RAB [RAB$B_RAC] = RAB$C_SEQ; ! Set sequential access
238 0299 2 RAB [RAB$B_KRF] = 0; ! Rewind primary key if indexed
239 0300 2 IF NOT $PASSRMS_OP ($REWIND (RAB=.RAB))
240 0301 2 THEN
241 0302 2 IF .RAB [RAB$L_STS] NEQ RMSS_IOP AND ! Inappropriate operation
242 0303 2 .RAB [RAB$L_STS] NEQ RMSS_EOF AND ! End of file
243 0304 2 .RAB [RAB$L_STS] NEQ RMSS_BOF ! Already at beginning of file
244 0305 2 THEN
245 0306 2 $PASSIO_ERROR (PASS_ERRDURREW); ! Error during REWRITE
246 0307 2
247 0308 2 !+
248 0309 2 ! The file is now rewound. If we got RMSS_IOP, there's nothing
249 0310 2 ! more we need to do. Otherwise, split on the file organization
250 0311 2 ! to either truncate the file or delete all records.
251 0312 2 !-
252 0313 2

```

```

253 0314 3      IF .RAB [RAB$L_STS] NEQ RMS$_IOP
254 0315 3      THEN
255 0316 4      BEGIN
256 0317 4      +
257 0318 4      | Split depending on file organization
258 0319 4      |
259 0320 4      | IF .FAB [FAB$B_ORG] EQL FAB$C_SEQ  ! Sequential
260 0321 4      | THEN
261 0322 5      | BEGIN
262 0323 6      | IF NOT $PASSRMS_OP ($GET (RAB=.RAB))  ! Read first record, if any
263 0324 5      | THEN
264 0325 6      |     BEGIN
265 0326 6      |     IF .RAB [RAB$L_STS] NEQ RMS$_EOF  ! End of file?
266 0327 6      |     THEN
267 0328 6      |     $PASSIO_ERROR (PASS_ERRDURREW); ! Error during REWRITE
268 0329 6      |     END
269 0330 5      | ELSE
270 0331 6      | BEGIN
271 0332 6      | +
272 0333 6      | | GET succeeded. Do a $TRUNCATE.
273 0334 6      | |
274 0335 6      | |
275 0336 7      | | IF NOT $PASSRMS_OP ($TRUNCATE (RAB=.RAB))
276 0337 6      | | THEN
277 0338 6      | | $PASSIO_ERROR (PASS_ERRDURREW); ! Error during REWRITE
278 0339 6      | |
279 0340 5      | | END;
280 0341 5      | |
281 0342 5      | | +
282 0343 5      | | | So far, all we've done is set the end-of-file mark to the
283 0344 5      | | | beginning of the file. The allocated space is still there.
284 0345 5      | | | Set FAB$V_TEF so that all unused allocated space will be
285 0346 5      | | | returned when the file is closed.
286 0347 5      | | |
287 0348 5      | | |
288 0349 5      | | | FAB [FAB$V_TEF] = 1;
289 0350 5      | | |
290 0351 5      | | | END
291 0352 4      | | ELSE
292 0353 5      | | BEGIN
293 0354 5      | | +
294 0355 5      | | | Relative and Indexed organizations. Get and Delete all
295 0356 5      | | | records.
296 0357 5      | | |
297 0358 5      | | |
298 0359 5      | | | +
299 0360 5      | | | | Do an initial $GET to see if the file is empty.
300 0361 5      | | | |
301 0362 5      | | | |
302 0363 6      | | | | IF NOT $PASSRMS_OP ($GET (RAB=.RAB))
303 0364 5      | | | | THEN
304 0365 6      | | | | BEGIN
305 0366 6      | | | | +
306 0367 6      | | | | | If file is empty, then don't do anything else.
307 0368 6      | | | | | Otherwise, we'll have to require that the user have
308 0369 6      | | | | | exclusive access to the file.
309 0370 6      | | | | |

```



```

310 0371 6
311 0372 6      IF .RAB [RAB$L_STS] NEQ RMSS_EOF
312 0373 6      THEN
313 0374 6          $PASSIO_ERROR (PASS_ERRDURREW); ! Error during REWRITE
314 0375 6      END
315 0376 5      ELSE
316 0377 6          BEGIN
317 0378 6              +
318 0379 6              | We can only scratch the data in the file if we have exclusive
319 0380 6              | access to it.  If not, give an error
320 0381 6              | -
321 0382 6
322 0383 6          IF NOT .FAB [FAB$V_NIL]
323 0384 6          THEN
324 0385 6              $PASSIO_ERROR (PASS_REWNOTALL); ! REWRITE not allowed
325 0386 6
326 0387 6          WHILE 1 DO
327 0388 7              BEGIN
328 0389 8                  IF NOT $PASSRMS_OP ($DELETE (RAB=.RAB))
329 0390 7                  THEN
330 0391 7                      $PASSIO_ERROR (PASS_ERRDURREW); ! Error during REWRITE
331 0392 8                  IF NOT $PASSRMS_OP ($GET (RAB=.RAB))
332 0393 7                  THEN
333 0394 8                      BEGIN
334 0395 8                          IF .RAB [RAB$L_STS] EQL RMSS_EOF
335 0396 8                          THEN
336 0397 8                              EXITLOOP
337 0398 8                          ELSE
338 0399 8                              $PASSIO_ERROR (PASS_ERRDURREW); ! Error during REWRITE
339 0400 7                              END;
340 0401 6                          END;
341 0402 5                      END;
342 0403 4                  END;
343 0404 3              END;
344 0405 3
345 0406 3          |
346 0407 3          | Rewind again to put us at the beginning.
347 0408 3          | -
348 0409 3
349 0410 3          $PASSRMS_OP ($REWIND (RAB=.RAB)); ! Ignore return status
350 0411 2          END;
351 0412 2
352 0413 2          +
353 0414 2          | Set Generation mode.  Reset buffer pointers.
354 0415 2          | Set "truncate-on-put" so that first $PUT will succeed to a magtape.
355 0416 2          | An RMS restriction prevents ordinary $PUTs from succeeding if the
356 0417 2          | magtape file is empty.  This bit will be cleared after the first
357 0418 2          | $PUT.  It is ignored for non-sequential $PUTs.
358 0419 2          | -
359 0420 2
360 0421 2          FCB [FCB$V_INSPECTION] = 0;
361 0422 2          FCB [FCB$V_GENERATION] = 1;
362 0423 2          FCB [FCB$A_RECORD_BEG] = .RAB [RAB$L_UBF];
363 0424 2          FCB [FCB$A_RECORD_CUR] = .FCB [FCB$A_RECORD_BEG];
364 0425 2          FCB [FCB$L_RECORD_LEN] = .RAB [RAB$W_USZ];
365 0426 2          FCB [FCB$A_RECORD_END] = .FCB [FCB$A_RECORD_BEG] + .FCB [FCB$L_RECORD_LEN];
366 0427 2          FCB [FCB$V_EOF] = -1;

```

```

: 367 0428 2 FCB [FCBSV_LOCATE] = 0;
: 368 0429 2 FCB [FCBSV_PARTIAL_LINE] = 0;
: 369 0430 2 RAB [RABSV_TPT] = T.
: 370 0431 2
: 371 0432 2 !+
: 372 0433 2 ! Define the file buffer
: 373 0434 2 ! Validate the status bits
: 374 0435 2 ! Indicate successful completion
: 375 0436 2 ! Unlock the file
: 376 0437 2 !-
: 377 0438 2
: 378 0439 2 PFV [PFVSV_EOF_DEFINED] = 1;
: 379 0440 2 PFV [PFVSV_VALID] = 1;
: 380 0441 2 PFV [PFVSV_DFB] = 0;
: 381 0442 2 FCB [FCBSL_STATUS] = 0;
: 382 0443 2 PFV [PFVSV_LOCK] = 0;
: 383 0444 2
: 384 0445 2 RETURN;
: 385 0446 2
: 386 0447 1 END;

```

! End of routine PASS\$REWRITE2

```

.TITLE PASS$REWRITE2 REWRITE procedure
.IDENT \1-004\

```

```

.EXTRN PASS$REWRITE2, PASS$IO_HANDLER
.EXTRN PASS$VALIDATE, PFV
.EXTRN PASS$OPEN_IMPLICIT
.EXTRN PASS$OPEN, PASS$SIGNAL
.EXTRN PASS$K_GENNOTALL
.EXTRN PASS$WRITELN, SYS$REWIND
.EXTRN PASS$K_ERRDURREW
.EXTRN SYS$GET, SYS$TRUNCATE
.EXTRN PASS$K_REWNOTALL
.EXTRN SYS$DELETE

```

```

.PSECT _PASS$CODE, NOWRT, SHR, PIC, 2

```

```

.OFFC 00000
: 5B 00000000G 00 9E 00002 MOVAB SYS$REWIND, R11
: 5A 00000000G 00 9E 00009 MOVAB PASS$SIGNAL, R10
: 59 00000000G 00 9E 00010 MOVAB SYS$GET, R9
: 5E 08 C2 00017 SUBL2 #8, SP
: 7E D4 0001A CLRL ERROR_ADDR
: 04 AE 7C 0001C CLRQ UNWIND_ACT
: 6D 01CE CF DE 0001F MOVAL 29$, (FP)
: 02 6C 91 00024 CMPB (AP), #2
: 04 1F 00027 BLSSU 1$
: 6E 08 AC D0 00029 MOVL ERROR, ERROR_ADDR
: 56 04 AC D0 0002D 1$: MOVL PFV, R6
: 08 AE 56 D0 00031 MOVL R6, PFV_ADDR
: 04 00000000G 00 16 00035 JSB PASS$VALIDATE, PFV
: 58 04 A6 9E 0003B MOVL #1, UNWIND_ACT
: 06 02 A8 E8 00043 MOVAB 4(R6), R8
: 00000000G 00 16 00047 BLBS 2(R8), 2$
: JSB PASS$OPEN_IMPLICIT

```

OB	68		1D	E0	0004D	2\$:	BBS	#29, (R8), 3\$	0249	
			04	DD	00051		PUSHL	#4	0251	
			56	DD	00053		PUSHL	R6		
0A	00000000G	00	02	FB	00055		CALLS	#2, PASS\$OPEN		
	FB	A7	04	E1	0005C	3\$:	BBC	#4, -8(FCB), 4\$	0257	
			7E	D4	00061		CLRL	-(SP)	0259	
		7E	00G	8F	9A	00063	MOVZBL	#PASSK GENNOTALL, -(SP)		
		6A		02	FB	00067	CALLS	#2, PASS\$SIGNAL		
				04	0006A		RET			
OD	FD	12	F8	A7	E9	0006B	4\$:	BLBC	-8(FCB), 5\$	0266
	A7	A7		04	E1	0006F		BBC	#4, -3(FCB), 5\$	
	E8	A7	EC	A7	D1	00074		CMPL	-20(FCB), -24(FCB)	0267
				06	1B	00079		BLEQU	5\$	
		52	00000000G	00	16	0007B		JSB	PASS\$WRITELN	0269
		A2	FC	A7	9E	00081	5\$:	MOVAB	-4(FCB), R2	0275
	01	A7		04	8A	00085		BICB2	#4, 1(R2)	
	06	A7		10	8A	00089		BICB2	#16, 6(FCB)	0281
		03	03	A2	E8	0008D		BLBS	3(R2), 7\$	0288
				0120	31	00091	6\$:	BRW	28\$	
		F8	00CA	C7	E8	00094	7\$:	BLBS	202(FCB), 6\$	
				1E	A7	94	00099	CLRB	30(FCB)	0298
				35	A7	94	0009C	CLRB	53(FCB)	0299
				57	DD	0009F	8\$:	PUSHL	FCB	0300
	6B			01	FB	000A1		CALLS	#1, SYSSREWIND	
	2E			50	E8	000A4		BLBS	\$\$STATUS, 10\$	
0001825A	8F			50	D1	000A7		CMPL	\$\$STATUS, #98906	
				04	12	000AE		BNEQ	9\$	
	EB	03		A2	E8	000B0		BLBS	3(R2), 8\$	
	1E			50	E8	000B4	9\$:	BLBS	\$\$STATUS, 10\$	
00018574	8F	08		A7	D1	000B7		CMPL	8(FCB), #99700	0302
				14	13	000BF		BEQL	10\$	
0001827A	8F	08		A7	D1	000C1		CMPL	8(FCB), #98938	0303
				0A	13	000C9		BEQL	10\$	
00018198	8F	08		A7	D1	000CB		CMPL	8(FCB), #98712	0304
				77	12	000D3		BNEQ	19\$	
00018574	8F	08		A7	D1	000D5	10\$:	CMPL	8(FCB), #99700	0314
				6B	13	000DD		BEQL	18\$	
			61	A7	95	000DF		TSTB	97(FCB)	0320
				46	12	000E2		BNEQ	16\$	
				57	DD	000E4	11\$:	PUSHL	FCB	0323
	69			01	FB	000E6		CALLS	#1, SYSSGET	
	1C			50	E8	000E9		BLBS	\$\$STATUS, 13\$	
0001825A	8F			50	D1	000EC		CMPL	\$\$STATUS, #98906	
				04	12	000F3		BNEQ	12\$	
	EB	03		A2	E8	000F5		BLBS	3(R2), 11\$	
	0C			50	E8	000F9	12\$:	BLBS	\$\$STATUS, 13\$	
0001827A	8F	08		A7	D1	000FC		CMPL	8(FCB), #98938	0326
				1E	13	00104		BEQL	15\$	
				44	11	00106		BRB	19\$	0328
				57	DD	00108	13\$:	PUSHL	FCB	0336
00000000G	00			01	FB	0010A		CALLS	#1, SYSSTRUNCATE	
	10			50	E8	00111		BLBS	\$\$STATUS, 15\$	
0001825A	8F			50	D1	00114		CMPL	\$\$STATUS, #98906	
				04	12	0011B		BNEQ	14\$	
	E7	03		A2	E8	0011D		BLBS	3(R2), 13\$	
	73			50	F9	00121	14\$:	BLBC	\$\$STATUS, 25\$	
4B	A7			10	B8	00124	15\$:	BISB2	#16, 75(FCB)	0349

				75	11	00128		BRB	27\$			0320
				57	DD	0012A	16\$:	PUSHL	FCB			0363
		69		01	FB	0012C		CALLS	#1, SYSSGET			
		1C		50	E8	0012F		BLBS	\$\$\$STATUS, 20\$			
		8F		50	D1	00132		CMPL	\$\$\$STATUS, #98906			
	0001825A			04	12	00139		BNEQ	17\$			
		EB	03	A2	E8	0013B		BLBS	3(R2), 16\$			
		OC		50	E8	0013F	17\$:	BLBS	\$\$\$STATUS, 20\$			
	0001827A			A7	D1	00142		CMPL	8(FCB), #98938			0372
		8F	08	53	13	0014A	18\$:	BEQL	27\$			
				49	11	0014C	19\$:	BRB	25\$			0374
	06	5B		05	E0	0014E	20\$:	BBS	#5, 91(FCB), 21\$			0383
		A7		8F	9A	00153		MOVZBL	#PASSK_REWNOTALL, -(SP)			0385
		7E	00G	42	11	00157		BRB	26\$			
				57	DD	00159	21\$:	PUSHL	FCB			0389
	00000000G	00		01	FB	0015B		CALLS	#1, SYSSDELETE			
		10		50	E8	00162		BLBS	\$\$\$STATUS, 23\$			
	0001825A			50	D1	00165		CMPL	\$\$\$STATUS, #98906			
		8F		04	12	0016C		BNEQ	22\$			
		E7	03	A2	E8	0016E		BLBS	3(R2), 21\$			
		22		50	E9	00172	22\$:	BLBC	\$\$\$STATUS, 25\$			
				57	DD	00175	23\$:	PUSHL	FCB			0392
		69		01	FB	00177		CALLS	#1, SYSSGET			
		DC		50	E8	0017A		BLBS	\$\$\$STATUS, 21\$			
	0001825A			50	D1	0017D		CMPL	\$\$\$STATUS, #98906			
		8F		04	12	00184		BNEQ	24\$			
		EB	03	A2	E8	00186		BLBS	3(R2), 23\$			
		CC		50	E8	0018A	24\$:	BLBS	\$\$\$STATUS, 21\$			
	0001827A			A7	D1	0018D		CMPL	8(FCB), #98938			0395
		8F	08	08	13	00195		BEQL	27\$			
		7E	00G	8F	9A	00197	25\$:	MOVZBL	#PASSK_ERRDURREW, -(SP)			0399
		6A		01	FB	0019B	26\$:	CALLS	#1, PASS\$SIGNAL			
					04	0019E		RET				
				57	DD	0019F	27\$:	PUSHL	FCB			0410
		6B		01	FB	001A1		CALLS	#1, SYSSREWIND			
		OD		50	E8	001A4		BLBS	\$\$\$STATUS, 28\$			
	0001825A			50	D1	001A7		CMPL	\$\$\$STATUS, #98906			
		8F		04	12	001AE		BNEQ	28\$			
		EB	03	A2	E8	001B0		BLBS	3(R2), 27\$			
		01		08	8A	001B4	28\$:	BICB2	#8, 1(R2)			0421
		01		10	88	001B8		BISB2	#16, 1(R2)			0422
		E8		A7	DO	001BC		MOVL	36(FCB), -24(FCB)			0423
		EC	24	A7	DO	001C1		MOVL	-24(FCB), -20(FCB)			0424
		F4	E8	A7	3C	001C6		MOVZWL	32(FCB), -12(FCB)			0425
		A7	20	A7	C1	001CB		ADDL3	-12(FCB), -24(FCB), -16(FCB)			0426
	FO	A7		F4	A7	001D2		BISB2	#32, 1(R2)			0427
		E8		A7	8F	AA	001D6	BICW2	#384, 1(R2)			0428
		01		A2	02	88	001DC	BISB2	#2, 4(FCB)			0430
		01	0180	A2	05	88	001E0	BISB2	#5, 2(R8)			0440
		04		A7	02	8A	001E4	BICB2	#2, 2(R8)			0441
		02		A8	A7	D4	001E8	CLRL	-4(FCB)			0442
		02		A8	8F	8A	001EB	BICB2	#128, 3(R8)			0443
					04	001F0		RET				0447
					0000	001F1	29\$:	.WORD	Save nothing			0195
		50		08	AC	DO	001F3	MOVL	8(AP), R0			
		50		04	A0	DO	001F7	MOVL	4(R0), R0			
				F4	A0	9F	001FB	PUSHAB	ERROR_ADDR			

PASSREWRITE2  
1-004

REWRITE procedure  
PASSREWRITE2 - REWRITE procedure

M 1  
16-Sep-1984 02:07:06  
14-Sep-1984 12:51:55

VAX-11 Bliss-32 V4.0-742  
[PASRTL.SRC]PASREWRIT.B32;1

Page 11  
(3)

PA  
1-

		F8	A0	9F	001FE	PUSHAB	UNWIND ACT	:
		FC	AC	9F	00201	PUSHAB	PFV_ADDR	:
			03	DD	00204	PUSHL	#3	:
			5E	DD	00206	PUSHL	SP	:
	7E	04	AC	7D	00208	MOVQ	4(AP), -(SP)	:
00000000G	00		03	FB	0020C	CALLS	#3, PASS\$IO_HANDLER	:
			04	00	00213	RET		:

; Routine Size. 532 bytes, Routine Base: \_PASS\$CODE + 0000

: 387 0448 1  
: 388 0449 1 !<BLF/PAGE>

PASS\$REWRITE2  
1-004

REWRITE procedure  
PASS\$REWRITE2 - REWRITE procedure

N 1  
16-Sep-1984 02:07:06  
14-Sep-1984 12:51:55

VAX-11 Bliss-32 V4.0-742  
[PASRTL.SPC]PAS\$REWRIT.B32;1

Page 12  
(4)

PA  
1-

: 390 0450 1 END  
: 391 0451 1  
: 392 0452 0 ELUDOM

! End of module PASS\$REWRITE2

PSECT SUMMARY

Name	Bytes	Attributes
_PASS\$CODE	532	NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	24	0	581	00:01.0
_\$255\$DUA28:[PASRTL.OBJ]PASLIB.L32;1	427	100	23	33	00:00.4

COMMAND QUALIFIERS

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

: Size: 532 code + 0 data bytes  
: Run Time: 00:13.1  
: Elapsed Time: 00:48.9  
: Lines/CPU Min: 2067  
: Lexemes/CPU-Min: 21736  
: Memory Used: 210 pages  
: Compilation Complete

PASREWR LIS PASIGNAL LIS PASSOR LIS  
PASRTRNCA LIS PASUNLCK LIS PASWR1B00 LIS  
PASVECTOR LIS PASWR1B1N LIS  
PASUNDEF1 LIS PASVALIDA LIS  
PASUM LIS PASWR1FF1 LIS  
PASTIME LIS PASUPDATE LIS PASWR1F1D1 LIS  
PASSTATUS LIS PASWR1F1D1 LIS  
PASUFB LIS PASWR1TCH1 LIS  
PASWR1TCH1 LIS PASWR1TCH1 LIS