


```

PPPPPPPP      AAAAAA      SSSSSSSS      PPPPPPPP      UU      UU      TTTTTTTTTT
PPPPPPPP      AAAAAA      SSSSSSSS      PPPPPPPP      UU      UU      TTTTTTTTTT
PP      PP      AA      AA      SS      PP      PP      UU      UU      TT
PP      PP      AA      AA      SS      PP      PP      UU      UU      TT
PP      PP      AA      AA      SS      PP      PP      UU      UU      TT
PP      PP      AA      AA      SS      PP      PP      UU      UU      TT
PPPPPPPP      AA      AA      SSSSSS      PPPPPPPP      UU      UU      TT
PPPPPPPP      AA      AA      SSSSSS      PPPPPPPP      UU      UU      TT
PP      AAAAAAAAAA      SS      PP      UU      UU      TT
PP      AAAAAAAAAA      SS      PP      UU      UU      TT
PP      AA      AA      SS      PP      UU      UU      TT
PP      AA      AA      SS      PP      UU      UU      TT
PP      AA      AA      SSSSSSSS      PP      UUUUUUUUUU      TT
PP      AA      AA      SSSSSSSS      PP      UUUUUUUUUU      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
LLLLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLLLL      IIIIII      SSSSSSSS

```

```

1 0001 0 MODULE PASSPUT ( %TITLE 'PUT procedure'
2 0002 0 IDENT = '1-001' ! File: PASPUT.B32 Edit: SBL1001
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 INFRMATION 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 PASSPUT, which implements the
36 0036 1 VAX-11 Pascal PUT 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 --
46 0046 1

```

PASSPUT
1-001

PUT procedure
Declarations

L 16
16-Sep-1984 01:51:30
14-Sep-1984 12:51:42

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

Page 2
(2)

```

: 48      0047 1 %SBTTL 'Declarations'
: 49      0048 1
: 50      0049 1 : PROLOGUE DEFINITIONS:
: 51      0050 1 :
: 52      0051 1
: 53      0052 1 REQUIRE 'RTLIN:PASPROLOG';           ! Linkages, externals, PSFCIs, structures
: 54      0116 1
: 55      0117 1
: 56      0118 1 : TABLE OF CONTENTS:
: 57      0119 1 :
: 58      0120 1
: 59      0121 1 FORWARD ROUTINE
: 60      0122 1   PASSPUT: NOVALUE;                 ! Do a PUT
: 61      0123 1
: 62      0124 1
: 63      0125 1 : MACROS:
: 64      0126 1
: 65      0127 1   NONE
: 66      0128 1
: 67      0129 1 : EQUATED SYMBOLS:
: 68      0130 1
: 69      0131 1   NONE
: 70      0132 1
: 71      0133 1 : FIELDS:
: 72      0134 1
: 73      0135 1   NONE
: 74      0136 1
: 75      0137 1 : OWN STORAGE:
: 76      0138 1
: 77      0139 1 :   NONE
```

```

79 0140 1 %SBTTL 'PASSPUT - PUT procedure'
80 0141 1 GLOBAL ROUTINE PASSPUT (           ! Do a PUT
81 0142 1   PFV: REF $PASSPFV_FILE_VARIABLE, ! File variable
82 0143 1   ERROR                               ! Error unwind address
83 0144 1   ): NOVALUE =
84 0145 1
85 0146 1 ++
86 0147 1 FUNCTIONAL DESCRIPTION:
87 0148 1
88 0149 1   PASSPUT implements the VAX-11 Pascal PUT procedure. It
89 0150 1   writes the current file buffer component to the file.
90 0151 1
91 0152 1 CALLING SEQUENCE:
92 0153 1
93 0154 1   CALL PASSPUT (PFV.mr.r [, ERROR.j.r])
94 0155 1
95 0156 1 FORMAL PARAMETERS:
96 0157 1
97 0158 1   PFV           - The Pascal File Variable (PFV) passed by reference.
98 0159 1   The structure of the PFV is defined in PASPFV.REQ.
99 0160 1
100 0161 1   ERROR        - Optional. If specified, the address to unwind to
101 0162 1   in case of an error.
102 0163 1
103 0164 1 IMPLICIT INPUTS:
104 0165 1
105 0166 1   NONE
106 0167 1
107 0168 1 IMPLICIT OUTPUTS:
108 0169 1
109 0170 1   NONE
110 0171 1
111 0172 1 ROUTINE VALUE:
112 0173 1
113 0174 1   NONE
114 0175 1
115 0176 1 SIDE EFFECTS:
116 0177 1
117 0178 1   Opens standard files INPUT or OUTPUT if not already open.
118 0179 1
119 0180 1 SIGNALLED ERRORS:
120 0181 1
121 0182 1   FILNOTGEN - file is not in Generation mode
122 0183 1   LINTOOLON - line too long
123 0184 1   LINVALEXC - LINELIMIT value exceeded
124 0185 1   KEYDUPNOT - key field duplication is not allowed
125 0186 1   ERRDURPUT - error during PUT
126 0187 1 --
127 0188 1
128 0189 2 BEGIN
129 0190 2
130 0191 2 LOCAL
131 0192 2   FCB: REF $PASSFCB_CONTROL_BLOCK, ! File control block
132 0193 2   PFV_ADDR: VOLATILE,           ! Enable argument
133 0194 2   UNWIND_ACT: VOLATILE,         ! Enable argument
134 0195 2   ERROR_ADDR: VOLATILE,        ! Enable argument
135 0196 2   STATUS;                       ! Status from $PUT

```

```

136 0197 2
137 0198 2
138 0199 2 BIND
139 0200 2 RAB = FCB: REF BLOCK [, BYTE]; ! RMS RAB
140 0201 2
141 0202 2 BUILTIN
142 0203 2 ACTUALCOUNT,
143 0204 2 TESTBITSS;
144 0205 2
145 0206 2 ENABLE
146 0207 2 PASS$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR); ! Enable error handler
147 0208 2
148 0209 2 IF ACTUALCOUNT () GEQU 2 ! Was ERROR specified?
149 0210 2 THEN
150 0211 2 ERROR_ADDR = .ERROR;
151 0212 2
152 0213 2 !+
153 0214 2 ! Set enable argument for PFV address.
154 0215 2 !-
155 0216 2 PFV_ADDR = PFV [PFV$R_PFV];
156 0217 2
157 0218 2 !+
158 0219 2 ! Validate and lock PFV.
159 0220 2 !-
160 0221 2
161 0222 2 PASS$VALIDATE_PFV (PFV [PFV$R_PFV]; FCB);
162 0223 2
163 0224 2 !+
164 0225 2 ! Set unwind action to unlock file.
165 0226 2 !-
166 0227 2
167 0228 2 UNWIND_ACT = PASS$UNWIND_UNLOCK;
168 0229 2
169 0230 2 !+
170 0231 2 ! Resolve lazy lookahead, if any. Also may open INPUT or OUTPUT.
171 0232 2 !-
172 0233 2
173 0234 2 IF NOT .PFV [PFV$V_VALID]
174 0235 2 THEN
175 0236 2 $PASS$LOOK_AHEAD (PFV [PFV$R_PFV], FCB [FCB$R_FCB]; FCB);
176 0237 2
177 0238 2 !+
178 0239 2 ! Verify that the file is open.
179 0240 2 !-
180 0241 2
181 0242 2 IF NOT .PFV [PFV$V_OPEN]
182 0243 2 THEN
183 0244 2 $PASS$IO_ERROR (PASS$_FILNOTOPE,0);
184 0245 2
185 0246 2 !+
186 0247 2 ! Verify that we are in Generation mode
187 0248 2 !-
188 0249 2
189 0250 2 IF NOT .FCB [FCB$V_GENERATION] AND NOT .FCB [FCB$V_INDEXED]
190 0251 2 THEN
191 0252 2 $PASS$IO_ERROR (PASS$_FILNOTGEN,0);
192 0253 2

```

```

193 0254 2  !+
194 0255 2  ! PUT from a textfile is different...
195 0256 2  !-
196 0257 2  !-
197 0258 2  IF .FCB [FCBSV_TEXT]
198 0259 2  THEN
199 0260 3  BEGIN
200 0261 3  !+
201 0262 3  ! Are we already at the end-of-line? If so, give an error.
202 0263 3  !-
203 0264 3  !-
204 0265 3  !-
205 0266 3  IF .FCB [FCBSA_RECORD_CUR] GEQA .FCB [FCBSA_RECORD_END]
206 0267 3  THEN
207 0268 4  $PASSIO_ERROR (PASS_LINTOOLON,1,1)
208 0269 3  ELSE
209 0270 4  BEGIN
210 0271 4  !+
211 0272 4  ! Move component to file and advance current position.
212 0273 4  !-
213 0274 4  CH$WCHAR_A (..PFV [PFV$A_BUFFER], FCB [FCBSA_RECORD_CUR]);
214 0275 3  END;
215 0276 3  END
216 0277 2  ELSE
217 0278 2  BEGIN
218 0279 2  !+
219 0280 2  ! Not a textfile, check and update the linelimit.
220 0281 2  ! Then call RMS to write the record.
221 0282 2  !-
222 0283 2  !-
223 0284 2  IF .FCB [FCBSL_LINELIMIT] GEQ 0 ! Limit in effect?
224 0285 2  THEN
225 0286 3  IF .FCB [FCBSL_LINELIMIT] EQL 0 ! Limit exceeded?
226 0287 3  THEN
227 0288 4  $PASSIO_ERROR (PASS_LINVALEXC,0)
228 0289 3  ELSE
229 0290 4  FCB [FCBSL_LINELIMIT] = .FCB [FCBSL_LINELIMIT] - 1;
230 0291 3  !-
231 0292 2  !+
232 0293 2  ! Set RSZ appropriately.
233 0294 2  !-
234 0295 2  !-
235 0296 2  IF .FCB [FCBSV_VARYING]
236 0297 2  THEN
237 0298 3  RAB [RABS$RSZ] = (..PFV [PFV$A_BUFFER])<0,16,0>
238 0299 2  ELSE
239 0300 3  RAB [RABS$RSZ] = .RAB [RABS$USZ];
240 0301 2  !-
241 0302 2  RAB [RABS$L_RBF] = .RAB [RABS$L_UBF];
242 0303 2  !-
243 0304 2  !+
244 0305 2  ! If the file was opened ACCESS:=KEYED, or the last thing we did
245 0306 2  ! was a LOCATE, set the access method to keyed, otherwise sequential.
246 0307 2  !-
247 0308 2  !-
248 0309 2  IF .FCB [FCBSV_KEYED] OR .FCB [FCBSV_LOCATE]
249 0310 2  THEN

```

```

250 0311 4          BEGIN
251 0312 4          RAB [RAB$B_RAC] = RAB$C_KEY;
252 0313 4          FCB [FCB$V_LOCATE] = 0;      ! Cancel locate
253 0314 4          END
254 0315 3          ELSE
255 0316 3          RAB [RAB$B_RAC] = RAB$C_SEQ;
256 0317 3
257 0318 3          STATUS = $PASSRMS_OP ($PUT (RAB=.RAB));
258 0319 3          IF NOT .STATUS
259 0320 3          THEN
260 0321 4          BEGIN
261 0322 4          IF .STATUS EQL RMSS_DUP
262 0323 4          THEN
263 0324 5          $PASSIO_ERROR (PASS_KEYDUPNOT) ! Key field duplication not allowed
264 0325 4          ELSE
265 0326 4          $PASSIO_ERROR (PASS_ERRDURPUT); ! Error during PUT
266 0327 3          END;
267 0328 3
268 0329 3          RAB [RAB$V_TPT] = 0;      ! Clear "truncate-on-put" which may have been
269 0330 3          ! set by a REWRITE.
270 0331 2          END;
271 0332 2
272 0333 2          !+
273 0334 2          !- Undefine file buffer
274 0335 2          !-
275 0336 2
276 0337 2          PFV [PFV$V_DFB] = 0;
277 0338 2
278 0339 2
279 0340 2          !+
280 0341 2          !- Indicate successful completion
281 0342 2          !- Unlock the file
282 0343 2          !-
283 0344 2
284 0345 2          FCB [FCB$L_STATUS] = 0;
285 0346 2          PFV [PFV$V_LOCK] = 0;
286 0347 2
287 0348 2          RETURN;
288 0349 2
289 0350 1          END;

```

! End of routine PASSPUT

```

.TITLE PASSPUT PUT procedure
.IDENT \1-001\

.EXTRN PASS$IO_HANDLER
.EXTRN PASS$VACIDATE PFV
.EXTRN PASS$LOOK_AHEAD
.EXTRN PASS$SIGNAL, PASSK_FILNOTUPE
.EXTRN PASSK_FILNOTGEN
.EXTRN PASSK_LINTOOLON
.EXTRN PASSK_LINVALEXC
.EXTRN SYSSPOT, PASSK_KEYDUPNOT
.EXTRN PASSK_ERRDURPUT

.PSECT _PASSCODE,NOWRT, SHR, PIC,2

```


			00DC	00000	.ENTRY	PASSPUT, Save R2,R3,R4,R6,R7	0141
	54	00000000G	00	9E 00002	MOVAB	PASS\$SIGNAL, R4	
	5E		08	C2 00009	SUBL2	#8, SP	
			7E	D4 0000C	CLRL	ERROR_ADDR	0189
		04	AE	7C 0000E	CLRQ	UNWIND_ACT	
	6D	00EB	CF	DE 00011	MOVAL	20\$, (FP)	
	02		6C	91 00016	CMPB	(AP), #2	0208
			04	1F 00019	BLSSU	1\$	
	6E	08	AC	D0 0001B	MOVL	ERROR, ERROR_ADDR	0210
	56	04	AC	D0 0001F	MOVL	PFV, R6	0216
08	AE		56	D0 00023	MOVL	R6, PFV_ADDR	
		00000000G	00	16 00027	JSB	PASS\$VACIDATE_PFV	0222
04	AE		01	D0 0002D	MOVL	#1, UNWIND_ACT	0228
	06	06	A6	E8 00031	BLBS	6(R6), 2\$	0234
		00000000G	C0	16 00035	JSB	PASS\$LOOK_AHEAD	0236
08	07	A6	05	E0 0003B	BBS	#5, 7(R6), 3\$	0242
			7E	D4 00040	CLRL	-(SP)	0244
	7E	00G	8F	9A 00042	MOVZBL	#PASSK_FILNOTOPE, -(SP)	
			42	11 00046	BRB	7\$	
	52	FC	A7	9E 00048	MOVAB	-4(FCB), R2	0250
0C	62		0C	E0 0004C	BBS	#12, (R2), 4\$	
08	62		16	E0 00050	BBS	#22, (R2), 4\$	
			7E	D4 00054	CLRL	-(SP)	0252
	7E	00G	8F	9A 00056	MOVZBL	#PASSK_FILNO*GEN, -(SP)	
			2E	11 0005A	BRB	7\$	
	1D	F8	A7	E9 0005C	BLBC	-8(FCB), 6\$	0258
F0	A7	EC	A7	D1 00060	CMPB	-20(FCB), -16(FCB)	0266
			0C	1F 00065	BLSSU	5\$	
			01	DD 00067	PUSHL	#1	0268
			01	DD 00069	PUSHL	#1	
	7E	00G	8F	9A 0006B	MOVZBL	#PASSK_LINTOOLON, -(SP)	
	64		03	FB 0006F	CALLS	#3, PASS\$SIGNAL	
			04	00072	RET		
EC	B7	00	B6	90 00073	MOVB	@0(R6), @-20(FCB)	0274
		EC	A7	D6 00078	INCL	-20(FCB)	
			76	11 0007B	BRB	19\$	0258
		E0	A7	D5 0007D	TSTL	-32(FCB)	0284
			0F	19 00080	BLSS	9\$	
			0A	12 00082	BNEQ	8\$	0286
			7E	D4 00084	CLRL	-(SP)	0288
	7E	00G	8F	9A 00086	MOVZBL	#PASSK_LINVALEXC, -(SP)	
	64		02	FB 0008A	CALLS	#2, PASS\$SIGNAL	
			04	0008D	RET		
		E0	A7	D7 0008E	DECL	-32(FCB)	0290
07	F8	A7	02	E1 00091	BBC	#2, -8(FCB), 10\$	0296
	22	A7	0C	B6 B0 00096	MOVW	@0(R6), 34(FCB)	0298
			05	11 0009B	BRB	11\$	
	22	A7	A7	B0 0009D	MOVW	32(FCB), 34(FCB)	0300
	28	A7	24	D0 000A2	MOVL	36(FCB), 40(FCB)	0302
04	62		02	E0 000A7	BBS	#2, (R2), 12\$	0309
		02	A2	E9 000AB	BLBC	2(R2), 13\$	
	1E	A7	01	90 000AF	MOVB	#1, 30(FCB)	0312
	02	A2	01	8A 000B3	BICB2	#1, 2(R2)	0313
			03	11 000B7	BRB	14\$	0309
		1E	A7	94 000B9	CLRB	30(FCB)	0316
			57	DD 000BC	PUSHL	FCB	0318
	00000000G	00	01	FB 000BE	CALLS	#1, SY\$PUT	

PASSPUT
1-001

PUT procedure
PASSPUT - PUT procedure

F 1
16-Sep-1984 01:51:30 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:51:42 [PASRTL.SRC]PASPUT.B32;1

Page 8
(3)

0001825A	27 8F		50 E8 000C5	BLBS	\$\$STATUS, 18\$	
			50 D1 000C8	CMPL	\$\$STATUS, #98906	
			04 12 000CF	BNEQ	15\$	
	E7	03	A2 E8 000D1	BLBS	3(R2), 14\$	
	17		50 E8 000D5	BLBS	STATUS, 18\$	0319
000184EC	8F		50 D1 000D8	CMPL	STATUS, #99564	0322
			06 12 000DF	BNEQ	16\$	
	7E	00G	8F 9A 000E1	MOVZBL	#PASSK_KEYDUPNOT, -(SP)	0324
			04 11 000E5	BRB	17\$	
	7E	00G	8F 9A 000E7	MOVZBL	#PASSK_ERRDURPUT, -(SP)	0326
	64		01 FB 000EB	CALLS	#1, PASS\$\$SIGNAL	
			04 000EE	RET		
	04 A7		02 8A 000EF	BICB2	#2, 4(FCB)	0329
	06 A6		02 8A 000F3	BICB2	#2, 6(R6)	0337
		D4	A7 D4 000F7	CLRL	-44(FCB)	0345
	07 A6	80	8F 8A 000FA	BICB2	#128, 7(R6)	0346
			04 000FF	RET		0350
			0000 00100	.WORD	Save nothing	0189
	50	08	AC D0 00102	MOVL	8(AP), R0	
	50	04	A0 D0 00106	MOVL	4(R0), R0	
		F4	A0 9F 0010A	PUSHAB	ERROR_ADDR	
		F8	A0 9F 0010D	PUSHAB	UNWIND_ACT	
		FC	A0 9F 00110	PUSHAB	PFV_ADDR	
			03 DD 00113	PUSHL	#3	
			5E DD 00115	PUSHL	SP	
	7E	04	AC 7D 00117	MOVQ	4(AP), -(SP)	
00000000G	00		03 FB 0011B	CALLS	#3, PASS\$\$IO_HANDLER	
			04 00122	RET		

; Routine Size: 291 bytes, Routine Base: _PASSCODE + 0000

; 290 0351 1
; 291 0352 1 !<BLF/PAGE>

PASSPUT
1-001

PUT procedure
PASSPUT - PUT procedure

G 1
16-Sep-1984 01:51:30 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:51:42 [PASRTI.SRC]PASPUT.B32;1

: 293 0353 1 END
: 294 0354 1
: 295 0355 0 ELUDOM

! End of module PASSPUT

PSECT SUMMARY

Name	Bytes	Attributes
_PASSCODE	291	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	14	0	581	00:01.0
_\$255\$DUA28:[PASRTL.OBJ]PASLIB.L32;1	427	95	22	33	00:00.4

COMMAND QUALIFIERS

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

: Size. 291 code + 0 data bytes
: Run Time: 00:08.6
: Elapsed Time: 00:26.2
: Lines/CPU Min: 2473
: Lexemes/CPU-Min: 18181
: Memory Used: 130 pages
: Compilation Complete

