

```

PPPPPPPPPPPP      AAAAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTT      LLL
PPPPPPPPPPPP      AAAAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTT      LLL
PPPPPPPPPPPP      AAAAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTT      LLL
PPP              PPP      AAA              AAA      SSS              RRR              RRR              TTT              LLL
PPP              PPP      AAA              AAA      SSS              RRR              RRR              TTT              LLL
PPP              PPP      AAA              AAA      SSS              RRR              RRR              TTT              LLL
PPP              PPP      AAA              AAA      SSS              RRR              RRR              TTT              LLL
PPP              PPP      AAA              AAA      SSS              RRR              RRR              TTT              LLL
PPP              PPP      AAA              AAA      SSS              RRR              RRR              TTT              LLL
PPPPPPPPPPPP      AAA              AAA              SSSSSSSSSS      RRRRRRRRRRRR      TTT              LLL
PPPPPPPPPPPP      AAA              AAA              SSSSSSSSSS      RRRRRRRRRRRR      TTT              LLL
PPPPPPPPPPPP      AAA              AAA              SSSSSSSSSS      RRRRRRRRRRRR      TTT              LLL
PPP              AAAAAAAAAAAAAAAAAA      SSS              RRR              RRR              TTT              LLL
PPP              AAAAAAAAAAAAAAAAAA      SSS              RRR              RRR              TTT              LLL
PPP              AAAAAAAAAAAAAAAAAA      SSS              RRR              RRR              TTT              LLL
PPP              AAA              AAA              SSS              RRR              RRR              TTT              LLL
PPP              AAA              AAA              SSS              RRR              RRR              TTT              LLL
PPP              AAA              AAA              SSS              RRR              RRR              TTT              LLL
PPP              AAA              AAA              SSS              RRR              RRR              TTT              LLL
PPP              AAA              AAA      SSSSSSSSSSSS      RRR              RRR              TTT      LLLLLLLLLLLLLLLLLL
PPP              AAA              AAA      SSSSSSSSSSSS      RRR              RRR              TTT      LLLLLLLLLLLLLLLLLL
PPP              AAA              AAA      SSSSSSSSSSSS      RRR              RRR              TTT      LLLLLLLLLLLLLLLLLL

```

2

Sym

PAS

PAS
PASPAS
PAS

PAS

PAS
PAS

PAS

PAS

PAS
PAS

PAS

PAS
PASPAS
PAS

PAS

PAS
PASPAS
PAS

PAS

PAS
PAS

PAS
PAS

PAS

PAS
PAS

PAS
PAS

PAS

PAS

FAS

PAS

PAS

FAS

PAS
PASPAS
PAS

PAS

PAS

PAS

PAS

PAS
PAS

PAS
PAS

PAS

PAS
PAS

PAS

```
PPPPPPPP      AAAAAA      SSSSSSSS      RRRRRRRR      EEEEEEEEEE      SSSSSSSS      EEEEEEEEEE      TTTTTTTTTT      222222
PPPPPPPP      AAAAAA      SSSSSSSS      RRRRRRRR      EEEEEEEEEE      SSSSSSSS      EEEEEEEEEE      TTTTTTTTTT      222222
PP      PP      AA      AA      SS      RR      RR      EE      SS      EE      TT      22      22
PP      PP      AA      AA      SS      RR      RR      EE      SS      EE      TT      22      22
PP      PP      AA      AA      SS      RR      RR      EE      SS      EE      TT      22      22
PPPPPPPP      AA      AA      SSSSSS      RRRRRRRR      EEEEEEEE      SSSSSS      EEEEEEEE      TT      22
PPPPPPPP      AA      AA      SSSSSS      RRRRRRRR      EEEEEEEE      SSSSSS      EEEEEEEE      TT      22
PP      AAAAAAAAAA      SS      RR      RR      EE      SS      EE      TT      22
PP      AAAAAAAAAA      SS      RR      RR      EE      SS      EE      TT      22
PP      AA      AA      SS      RR      RR      EE      SS      EE      TT      22
PP      AA      AA      SSSSSSSS      RR      RR      EEEEEEEEEE      SSSSSSSS      EEEEEEEEEE      TT      2222222222
PP      AA      AA      SSSSSSSS      RR      RR      EEEEEEEEEE      SSSSSSSS      EEEEEEEEEE      TT      2222222222
...

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      II      SS
LLLLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLLLL      IIIIII      SSSSSSSS
```



```
1 0001 0 MODULE PAS$RESET2 ( %TITLE 'RESET procedure'
2 0002 0 IDENT = '1-002' ! File: PASRESET2.B32 Edit: SBL1002
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: Pascal Language Support
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 This module contains PAS$RESET2, which implements the
36 0036 1 VAX-11 Pascal RESET 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 - Only do $REWIND on file oriented devices. SBL 29-July-1982
46 0046 1 --
47 0047 1
```

PASSRESET2
1-002

RESET procedure
Declarations

J 15
16-Sep-1984 02:05:23
14-Sep-1984 12:51:54

VAX-11 Bliss-32 V4.0-742
[PASRTL.SRC]PASRESET2.B32;1

Page 2
(2)

```

49      0048 1 %SBTTL 'Declarations'
50      0049 1
51      0050 1 PROLOGUE DEFINITIONS:
52      0051 1
53      0052 1
54      0053 1 REQUIRE 'RTLIN:PASPROLOG';           ! Externals, linkages, PSECTs, structures
55      0117 1
56      0118 1
57      0119 1 TABLE OF CONTENTS:
58      0120 1
59      0121 1
60      0122 1 FORWARD ROUTINE
61      0123 1     PASSRESET2: NOVALUE;             ! Do a RESET
62      0124 1
63      0125 1
64      0126 1 MACROS:
65      0127 1
66      0128 1     NONE
67      0129 1
68      0130 1 EQUATED SYMBOLS:
69      0131 1
70      0132 1     NONE
71      0133 1
72      0134 1 FIELDS:
73      0135 1
74      0136 1     NONE
75      0137 1
76      0138 1 OWN STORAGE:
77      0139 1
78      0140 1     NONE
```


PASSRESET2
1-002

RESET procedure
PASSRESET2 - RESET procedure

K 15
16-Sep-1984 02:05:23
14-Sep-1984 12:51:54

VAX-11 Bliss-32 V4.0-742
[PASRTL.SRC]PASSRESET2.B32;1

Page 3
(3)

```

80 0141 1 %SBTTL 'PASSRESET2 - RESET procedure'
81 0142 1 GLOBAL ROUTINE PASSRESET2 (
82 0143 1     PFV: REF $PASSPFV_FILE_VARIABLE,
83 0144 1     ERROR
84 0145 1 ): NOVALUE =
85 0146 1
86 0147 1 ++
87 0148 1 FUNCTIONAL DESCRIPTION:
88 0149 1
89 0150 1     PASSRESET2 implements the VAX-11 Pascal RESET procedure. It
90 0151 1     opens the file if not already open, rewinds to the beginning of
91 0152 1     the file, enters Inspection mode and does a GET.
92 0153 1
93 0154 1 CALLING SEQUENCE:
94 0155 1
95 0156 1     CALL PASSRESET2 (PFV.mr.r)
96 0157 1
97 0158 1 FORMAL PARAMETERS:
98 0159 1
99 0160 1     PFV           - The Pascal File Variable (PFV) passed by reference.
100 0161 1                The structure of the PFV is defined in PASPFV.REQ.
101 0162 1
102 0163 1     ERROR       - Optional. If specified, the address to unwind to
103 0164 1                in case of an error.
104 0165 1
105 0166 1 IMPLICIT INPUTS:
106 0167 1
107 0168 1     NONE
108 0169 1
109 0170 1 IMPLICIT OUTPUTS:
110 0171 1
111 0172 1     NONE
112 0173 1
113 0174 1 ROUTINE VALUE:
114 0175 1
115 0176 1     NONE
116 0177 1
117 0178 1 SIDE EFFECTS:
118 0179 1
119 0180 1     Throws away partial line contents, if any.
120 0181 1     Positions to beginning of file.
121 0182 1     Switches to Inspection mode.
122 0183 1     Does a GET.
123 0184 1
124 0185 1 SIGNALLED ERRORS:
125 0186 1
126 0187 1     ERRDURRES - Error during RESET
127 0188 1     RESNOTALL - RESET not allowed on unopened internal file
128 0189 1     INSNOTALL - Inspection mode not allowed for WRITEONLY file
129 0190 1
130 0191 1 --
131 0192 1
132 0193 2 BEGIN
133 0194 2
134 0195 2 LOCAL
135 0196 2     FCB: REF $PASSFCB_CONTROL_BLOCK,
136 0197 2     PFV_ADDR: VOLATILE,
```

```

! Do a RESET
! File variable
! Error unwind address
```

```

! File control block
! Enable argument
```

PASSRESET2
1-002

RESET procedure
PASSRESET2 - RESET procedure

L 15
16-Sep-1984 02:05:23 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:51:54 [PASRTL.SRC]PASSRESET2.B32;1

Page 4
(3)

```

137 0198 2      UNWIND_ACT: VOLATILE,      ! Enable argument
138 0199 2      ERROR_ADDR: VOLATILE;      ! Enable argument
139 0200 2
140 0201 2      BIND
141 0202 2      RAB = FCB: REF BLOCK [, BYTE];      ! RAB is also FCB address
142 0203 2
143 0204 2      BUILTIN
144 0205 2      ACTUALCOUNT;
145 0206 2
146 0207 2      ENABLE
147 0208 2      PASS$IO_HANDLER (PFV_ADDR, UNWIND_ACT, ERROR_ADDR);      ! Enable error handler
148 0209 2
149 0210 2      IF ACTUALCOUNT () GEQU 2
150 0211 2      THEN
151 0212 2      ERROR_ADDR = .ERROR;      ! Set unwind address
152 0213 2
153 0214 2      !+
154 0215 2      ! Set enable argument for PFV address.
155 0216 2      !-
156 0217 2
157 0218 2      PFV_ADDR = PFV [PFV$R_PFV];
158 0219 2
159 0220 2      !+
160 0221 2      ! Validate and lock PFV.
161 0222 2      !-
162 0223 2
163 0224 2      PASS$VALIDATE_PFV (PFV [PFV$R_PFV]; FCB);
164 0225 2
165 0226 2      !+
166 0227 2      ! Set unwind action to unlock file.
167 0228 2      !-
168 0229 2
169 0230 2      UNWIND_ACT = PASS$K_UNWIND_UNLOCK;
170 0231 2
171 0232 2      !+
172 0233 2      ! Open file if it should be implicitly opened.
173 0234 2      !-
174 0235 2
175 0236 2      IF NOT .PFV [PFV$V_VALID]
176 0237 2      THEN
177 0238 2      PASS$OPEN_IMPLICIT (PFV [PFV$R_PFV], FCB [FCB$R_FCB]; FCB);
178 0239 2
179 0240 2      !+
180 0241 2      ! See if the file is open. If not, open it.
181 0242 2      !-
182 0243 2
183 0244 2      IF NOT .PFV [PFV$V_OPEN]      ! Not open
184 0245 2      THEN
185 0246 2      BEGIN
186 0247 2      LOCAL
187 0248 2      PFD: REF $PASS$PFD_FILE_DESCRIPTOR;      ! File descriptor
188 0249 2
189 0250 2      !+
190 0251 2      ! If PFD address is relative, make it absolute.
191 0252 2      !-
192 0253 2
193 0254 2      IF .PFV [PFV$V_RELPFD]
```



```
194 0255 3      THEN
195 0256 4      BEGIN
196 0257 4      PFV [PFV$A_PFD] = .PFV [PFV$A_PFD] + PFV [PFV$R_PFV];
197 0258 4      PFV [PFV$V_REL_PFD] = 0;
198 0259 4      END;
199 0260
200 0261 3      PFD = .PFV [PFV$A_PFD]; ! Get PFD address
201 0262 3
202 0263 3      IF .PFD [PFD$V_EXTERN] ! Is it an external file?
203 0264 3      THEN
204 0265 3          PASS$OPEN (PFV [PFV$R_PFV], PASS$HISTORY_OLD; FCB)
205 0266 3      ELSE
206 0267 3          $PASSIO_ERROR (PASS$_RESNOTALL,0); ! RESET not allowed
207 0268 3      END;
208 0269
209 0270 3      !+
210 0271 3      ! If the file has the NOREAD attribute, signal an error.
211 0272 3      !-
212 0273 3
213 0274 3      IF .FCB [FCB$V_NOREAD]
214 0275 3      THEN
215 0276 3          $PASSIO_ERROR (PASS$_INSNOTALL,0); ! Inspection mode not allowed
216 0277 3
217 0278 3      !+
218 0279 3      ! If a textfile, and if in Generation mode, and if the buffer
219 0280 3      ! is not empty, do a WRITELN.
220 0281 3      !-
221 0282 3
222 0283 3      IF .FCB [FCB$V_TEXT] AND .FCB [FCB$V_GENERATION] AND
223 0284 3          (.FCB [FCB$A_RECORD_CUR] GT .FCB [FCB$A_RECORD_BEG])
224 0285 3      THEN
225 0286 3          PASS$WRITELN (PFV [PFV$R_PFV], FCB [FCB$R_FCB]);
226 0287 3
227 0288 3      !+
228 0289 3      ! Cancel any lazy lookahead in progress.
229 0290 3      !-
230 0291 3
231 0292 3      FCB [FCB$V_LAZY] = 0;
232 0293 3
233 0294 3      !+
234 0295 3      ! Is the device a file oriented device? If so, proceed with rewind.
235 0296 3      !-
236 0297 3
237 0298 3      IF .FCB [FCB$V_FOD] ! File Oriented Device
238 0299 3      THEN
239 0300 3          BEGIN
240 0301 3              RAB [RAB$B_RAC] = RAB$C_SEQ; ! Set sequential access
241 0302 3              RAB [RAB$V_NLK] = 0; ! Cancel previous no-lock
242 0303 3              RAB [RAB$B_KRF] = 0; ! Rewind primary key if indexed
243 0304 3              IF NOT $PASSRMS_OP ($REWIND (RAB=.RAB))
244 0305 3              THEN
245 0306 3                  IF .RAB [RAB$L_STS] NEQ RMSS$_IOP AND ! Inappropriate operation
246 0307 3                      .RAB [RAB$L_STS] NEQ RMSS$_EOF AND ! End of file
247 0308 3                      .RAB [RAB$L_STS] NEQ RMSS$_BOF ! Already at beginning of file
248 0309 3                  THEN
249 0310 3                      $PASSIO_ERROR (PASS$_ERRDURRES); ! Error during RESET
250 0311 3
```

PASSRESET2
1-002

RESET procedure
PASSRESET2 - RESET procedure

N 15
16-Sep-1984 02:05:23
14-Sep-1984 12:51:54

VAX-11 Bliss-32 V4.0-742
[PASRTL.SRC]PASSRESET2.B32;1

Page 6
(3)

```

251 0312 2      END;
252 0313 2
253 0314 2      !+
254 0315 2      ! Set Inspection mode.
255 0316 2      !-
256 0317 2
257 0318 2      FCB [FCB$V_INSPECTION] = 1;
258 0319 2      FCB [FCB$V_GENERATION] = 0;
259 0320 2      FCB [FCB$V_EOF] = 0;
260 0321 2      FCB [FCB$V_LOCATE] = 0;
261 0322 2      FCB [FCB$V_PARTIAL_LINE] = 0;
262 0323 2
263 0324 2      !+
264 0325 2      ! File is now rewound, do a GET (or set lazy-lookahead if textfile).
265 0326 2      !-
266 0327 2
267 0328 2      IF .FCB [FCB$V_TEXT]
268 0329 2      THEN
269 0330 2          BEGIN
270 0331 2              FCB [FCB$V_LAZY] = 1;      ! Lazy lookahead in progress
271 0332 2              PFV [PFV$V_VALID] = 0;    ! Buffer is not valid
272 0333 2              FCB [FCB$L_RECORD_NUMBER] = 0; ! Reset record number
273 0334 2          END
274 0335 2      ELSE
275 0336 2          PASS$GET (PFV [PFV$R_PFV], FCB [FCB$R_FCB]);
276 0337 2
277 0338 2      !+
278 0339 2      ! Indicate successful completion
279 0340 2      ! Unlock the file
280 0341 2      !-
281 0342 2
282 0343 2      FCB [FCB$L_STATUS] = 0;
283 0344 2      PFV [PFV$V_LOCK] = 0;
284 0345 2
285 0346 2      RETURN;
286 0347 2
287 0348 1      END;
```

! End of routine PASSRESET2

.TITLE PASSRESET2 RESET procedure
.IDENT \1-002\

.EXTRN PASSRESET2, PASS\$IO_HANDLER
.EXTRN PASS\$VALIDATE_PFV
.EXTRN PASS\$OPEN_IMPLICIT
.EXTRN PASS\$OPEN, PASS\$SIGNAL
.EXTRN PASSK_RESNOTALL
.EXTRN PASSK_INSNOTALL
.EXTRN PASS\$WRITELN, SYSSREWIND
.EXTRN PASSK_ERRDURRES
.EXTRN PASS\$GET

.PSECT _PASSCODE, NOWRT, SHR, PIC, 2

.ENTRY PASSRESET2, Save R2,R3,R4,R5,R6,R7,R8,R9
MOVAB PASS\$SIGNAL, R9
SUBL2 #8, SP

59 00000000G 00 03FC 00000
5E 08 9E 00002
08 C2 00009

: 0142
:
:

			7E	D4	0000C	CLRL	ERROR_ADDR	: 0193
		04	AE	7C	0000E	CLRQ	UNWIND_ACT	:
	6D	00FF	CF	DE	00011	MOVAL	14\$, (FP)	:
	02		6C	91	00016	CMPB	(AP), #2	: 0210
			04	1F	00019	BLSSU	1\$	:
	6E	08	AC	D0	0001B	MOVL	ERROR, ERROR_ADDR	: 0212
	56	04	AC	D0	0001F	MOVL	PFV, R6	: 0218
	08		56	D0	00023	MOVL	R6, PFV_ADDR	:
		00000000G	00	16	00027	JSB	PASS\$VALIDATE_PFV	: 0224
	04		01	D0	0002D	MOVL	#1, UNWIND_ACT	: 0230
			A6	9E	00031	MOVAB	4(R6), R8	: 0236
	58	04	A8	E8	00035	BLBS	2(R8), 2\$	:
	06	02	00	16	00039	JSB	PASS\$OPEN_IMPLICIT	: 0238
2A		00000000G	1D	E0	0003F	BBS	#29, (R8), 5\$	: 0244
08			1C	E1	00043	BBC	#28, (R8), 3\$	: 0254
	08		56	C0	00047	ADDL2	R6, 8(R6)	: 0257
	03		10	8A	0004B	BICB2	#16, 3(R8)	: 0258
			A6	D0	0004F	MOVL	8(R6), PFD	: 0261
0D	04		05	E1	00053	BBC	#5, 4(PFD), 4\$	: 0263
			03	DD	00058	PUSHL	R6	: 0265
			56	DD	0005A	CALLS	#2, PASS\$OPEN	:
	00000000G	00	02	FB	0005C	BRB	5\$	:
			08	11	00063	CLRL	-(SP)	: 0267
			7E	D4	00065	MOVZBL	#PASSK_RESNOTALL, -(SP)	:
		00G	8F	9A	00067	BRB	6\$	:
0A	F8		0B	11	0006B	BBC	#3, -8(FCB), 7\$	: 0274
			03	E1	0006D	CLRL	-(SP)	: 0276
			7E	D4	00072	MOVZBL	#PASSK_INSNOTALL, -(SP)	:
		00G	8F	9A	00074	CALLS	#2, PASS\$SIGNAL	:
			02	FB	00078	RET		:
			04	0007B	BLBC	-8(FCB), 8\$	: 0283	
0D	FD		A7	E9	0007C	BBC	#4, -3(FCB), 8\$	: 0284
	E8		A7	D1	00085	CMP	-20(FCB), -24(FCB)	: 0286
			06	1B	0008A	BLEQU	8\$	: 0292
		00000000G	00	16	0008C	JSB	PASS\$WRITELN	: 0298
		FC	A7	9E	00092	MOVAB	-4(FCB), R2	: 0302
	01		04	8A	00096	BICB2	#4, 1(R2)	: 0303
			A2	E9	0009A	BLBC	3(R2), 11\$	: 0304
			A7	94	0009E	CLRB	30(FCB)	: 0305
	06		10	8A	000A1	BICB2	#16, 6(FCB)	:
			A7	94	000A5	CLRB	53(FCB)	:
		35	57	DD	000A8	PUSHL	FCB	:
	00000000G	00	01	FB	000AA	CALLS	#1, SYSS\$REWIND	:
		36	50	E8	000B1	BLBS	\$\$STATUS, 11\$	:
	0001825A	8F	50	D1	000B4	CMP	\$\$STATUS, #98906	:
			04	12	000BB	BNEQ	10\$	:
			A2	E8	000BD	BLBS	3(R2), 9\$	:
		03	50	E8	000C1	BLBS	\$\$STATUS, 11\$	: 0307
	00018574	8F	A7	D1	000C4	CMP	8(FCB), #99700	: 0308
			1C	13	000CC	BEQL	11\$	: 0309
	0001827A	8F	A7	D1	000CE	CMP	8(FCB), #98938	:
			12	13	000D6	BEQL	11\$	:
	00018198	8F	A7	D1	000D8	CMP	8(FCB), #98712	:
			08	13	000E0	BEQL	11\$	: 0311
			8F	9A	000E2	MOVZBL	#PASSK_ERRDURRES, -(SP)	:
		00G	01	FB	000E6	CALLS	#1, PASS\$SIGNAL	:

PASSRESET2
1-002

RESET procedure
PASSRESET2 - RESET procedure

C 16
16-Sep-1984 02:05:23
14-Sep-1984 12:51:54

VAX-11 Bliss-32 V4.0-742
[PASRTL.SRC]PASRESET2.B32;1

Page 8
(3)

01	A2		08	04	000E9	RET		
01	A2	01B0	8F	88	000EA	BISB2	#8, 1(R2)	0318
	0D	F8	A7	AA	000EE	BICW2	#432, 1(R2)	0322
01	A2		04	E9	000F4	BLBC	-8(FCB), 12\$	0328
02	A8		01	88	000F8	BISB2	#4, 1(R2)	0331
		C8	07	8A	000FC	BICB2	#1, 2(R8)	0332
			A7	D4	00100	CLRL	-56(FCB)	0333
			06	11	00103	BRB	13\$	0328
		00000000G	00	16	00105	JSB	PASS\$GET	0336
		D4	A7	D4	0010B	CLRL	-44(FCB)	0343
03	A8	80	8F	8A	0010E	BICB2	#128, 3(R8)	0344
			04	04	00113	RET		0348
				0000	00114	.WORD	Save nothing	0193
	50	08	AC	D0	00116	MOVL	8(AP), R0	
	50	04	A0	DC	0011A	MOVL	4(R0), R0	
		F4	A0	9F	0011E	PUSHAB	ERROR_ADDR	
		F8	A0	9F	00121	PUSHAB	UNWIND_ACT	
		FC	A0	9F	00124	PUSHAB	PFV_ADDR	
			03	DD	00127	PUSHL	#3	
			5E	DD	00129	PUSHL	SP	
	7E	04	AC	7D	0012B	MOVQ	4(AP), -(SP)	
00000000G	00		03	FB	0012F	CALLS	#3, PASS\$IO_HANDLER	
			04	04	00136	RET		

; Routine Size: 311 bytes, Routine Base: _PASS\$CODE + 0000

; 288 0349 1
; 289 0350 1 !<BLF/PAGE>

PASSRESET2
1-002

RESET procedure
PASSRESET2 - RESET procedure

D 16
16-Sep-1984 02:05:23 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:51:54 [PASRTL.SRC]PASRESET2.B32;1

Page 9
(4)

: 291 0351 1 END
: 292 0352 1
: 293 0353 0 ELUDOM

! End of module PASSRESET2

PSECT SUMMARY

Name	Bytes	Attributes
_PASSCODE	311	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	13	0	581	00:01.0
-\$255\$DUA28:[PASRTL.OBJ]PASLIB.L32;1	427	122	28	33	00:00.4

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS\$:PASRESET2/OBJ=OBJ\$:PASRESET2 MSRC\$:PASRESET2/UPDATE=(ENH\$:PASRESET2
:)

: Size: 311 code + 0 data bytes
: Run Time: 00:08.4
: Elapsed Time: 00:28.7
: Lines/CPU Min: 2536
: Lexemes/CPU-Min: 17446
: Memory Used: 136 pages
: Compilation Complete

0296 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

