

```

EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEEEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTT
EEEEEEEEEEEEEEEEEE DDDDDDDDDDDDDD TTT

```

EXE

Mod

EDT

ED

ED
EDED
EDED
EDED
ED

ED

ED

ED

ED

ED
EDSYN
LBA

110

```

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

```



```
0001 0 %TITLE 'EDT$PRISTOK' - test for particular token'
0002 0 MODULE EDT$PRISTOK ( ! Test for particular token
0003 0 IDENT = 'V04-000' ! File: PRISTOK.BLI Edit: JBS1004
0004 0 ) =
0005 1 BEGIN
0006 1
0007 1 *****
0008 1 *
0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0011 1 * ALL RIGHTS RESERVED.
0012 1 *
0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0018 1 * TRANSFERRED.
0019 1 *
0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0022 1 * CORPORATION.
0023 1 *
0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0026 1 *
0027 1 *
0028 1 *****
0029 1
0030 1
0031 1 ++
0032 1 FACILITY: EDT -- The DEC Standard Editor
0033 1
0034 1 ABSTRACT:
0035 1
0036 1 Test for a particular token.
0037 1
0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0039 1
0040 1 AUTHOR: Bob Kushlis, CREATION DATE: December 12, 1978
0041 1
0042 1 MODIFIED BY:
0043 1
0044 1 1-001 - Original. DJS 25-Feb-1981. This module was created by
0045 1 extracting routine IS_TOKEN from module PARSER.
0046 1 1-002 - Regularize headers. JBS 12-Mar-1981
0047 1 1-003 - Use the ASSERT macro. JBS 01-Jun-1981
0048 1 1-004 - Improve the appearance of the listing. JBS 17-Jun-1983
0049 1 --
0050 1
```

EDT\$PRISTOK
V04-000

EDT\$PRISTOK - test for particular token
Declarations

I 16
16-Sep-1984 01:18:30
14-Sep-1984 12:24:09

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]PRISTOK.BLI;1 Page 2
(2)

```

52 0051 1 %SBTTL 'Declarations'
53 0052 1
54 0053 1 TABLE OF CONTENTS:
55 0054 1
56 0055 1
57 0056 1 REQUIRE 'EDTSRC:TRAROUNAM';
58 0495 1
59 0496 1 FORWARD ROUTINE
60 0497 1 EDT$SPA_TSTTOK;
61 0498 1
62 0499 1
63 0500 1 INCLUDE FILES:
64 0501 1
65 0502 1
66 0503 1 REQUIRE 'EDTSRC:EDTREQ';
67 0638 1
68 0639 1 REQUIRE 'EDTSRC:PARLITS';
69 0923 1
70 0924 1 REQUIRE 'EDTSRC:PARTOKDEF';
71 1064 1
72 1065 1 REQUIRE 'EDTSRC:PARTOKTBL';
73 1078 1
74 1079 1
75 1080 1 MACROS:
76 1081 1
77 1082 1 NONE
78 1083 1
79 1084 1 EQUATED SYMBOLS:
80 1085 1
81 1086 1 NONE
82 1087 1
83 1088 1 OWN STORAGE:
84 1089 1
85 1090 1 NONE
86 1091 1
87 1092 1 EXTERNAL REFERENCES:
88 1093 1
89 1094 1 In the routine
```



```

91 1095 1 %SBTTL 'EDT$PA_TSTTOK - test for particular token'
92 1096 1
93 1097 1 GLOBAL ROUTINE EDT$PA_TSTTOK (           ! Test for a particular token
94 1098 1     TOK_NUM                             ! Number of the model token
95 1099 1     ) =
96 1100 1
97 1101 1
98 1102 1 ++
99 1103 1 FUNCTIONAL DESCRIPTION:
100 1104 1     This routine compares the current token as described by
101 1105 1     EDT$G_PA_TOKCLASS , EDT$A_PA_CURTOK and
102 1106 1     EDT$G_PA_CURTOKLEN to the token specified by the input parameter.
103 1107 1
104 1108 1     When EDT$G_PA_TOKCLASS is CL_NUM or CL_STRING the token will match only
105 1109 1     T_NUM and T_STR respectively. Otherwise, compare the token to the token_table
106 1110 1     entry specified by TOK_NUM. EDT$G_PA_PCEN will be on if the token
107 1111 1     started with a '%' and in that case, check the bit to see if an optional percent
108 1112 1     is allowed.
109 1113 1
110 1114 1 FORMAL PARAMETERS:
111 1115 1
112 1116 1     TOK_NUM             The number of the model token
113 1117 1
114 1118 1 IMPLICIT INPUTS:
115 1119 1
116 1120 1     EDT$A_PA_CURTOK
117 1121 1     EDT$G_PA_CURTOKLEN
118 1122 1     EDT$G_PA_PCEN
119 1123 1     EDT$G_PA_TOKCLASS
120 1124 1
121 1125 1 IMPLICIT OUTPUTS:
122 1126 1
123 1127 1     NONE
124 1128 1
125 1129 1 ROUTINE VALUE:
126 1130 1
127 1131 1     1 = token matches
128 1132 1     0 = token does not match
129 1133 1
130 1134 1 SIDE EFFECTS:
131 1135 1
132 1136 1     NONE
133 1137 1
134 1138 1 --
135 1139 1
136 1140 2 BEGIN
137 1141 2
138 1142 2 EXTERNAL
139 1143 2     EDT$A_PA_CURTOK,           ! start of the current token
140 1144 2     EDT$G_PA_CURTOKLEN,    ! length of current token
141 1145 2     EDT$G_PA_PCEN,         ! Did the keyword contain a percent?
142 1146 2     EDT$G_PA_TOKCLASS;    ! class of current token
143 1147 2
144 1148 2 LOCAL
145 1149 2     TOK;                     ! Index into the token table of the token we are looking for
146 1150 2
147 1151 2 !+
```

```
148 1152 2 ! Case on the token class.
149 1153 2 !-
150 1154 2
151 1155 2 CASE .EDT$$G_PA_TOKCLASS FROM CL_NAME TO CL_STRING OF
152 1156 2 SET
153 1157 2
154 1158 2 [CL_NUMBER] :
155 1159 2
156 1160 2 IF (.TOK_NUM EQL T_NUM) THEN RETURN (1) ELSE RETURN (0);
157 1161 2
158 1162 2 [CL_STRING] :
159 1163 2
160 1164 2 IF (.TOK_NUM EQL T_STR) THEN RETURN (1) ELSE RETURN (0);
161 1165 2
162 1166 2 [CL_NAME, CL_SPECIAL] :
163 1167 2 BEGIN
164 1168 2 !+
165 1169 3 ! Get index of the token.
166 1170 3 !-
167 1171 3 TOK = .TOK_INDX [.TOK_NUM - 1];
168 1172 4 BEGIN
169 1173 4 !+
170 1174 4 ! Bind the token descriptor.
171 1175 4 !-
172 1176 4
173 1177 4 BIND
174 1178 4 TOKEN = TOKEN_TABLE [.TOK] : BLOCK [2, BYTE];
175 1179 4
176 1180 4 !+
177 1181 4 ! Check for optional PERCENT.
178 1182 4 !-
179 1183 4
180 1184 4 IF (.EDT$$G_PA_PCEN AND ( NOT .TOKEN [PAR_OPT_PERCENT])) THEN RETURN (0);
181 1185 4
182 1186 4 !+
183 1187 4 ! Compare the token.
184 1188 4
185 1189 4
186 1190 5 IF ((.EDT$$G_PA_CURTOKLEN GEQ .TOKEN [PAR_MIN_LENGTH]) AND !
187 1191 5 (.EDT$$G_PA_CURTOKLEN LEQ .TOKEN [PAR_MAX_LENGTH]))
188 1192 4 THEN
189 1193 4
190 1194 4 IF CH$EQL (.EDT$$G_PA_CURTOKLEN, .EDT$$A_PA_CURTOK, .EDT$$G_PA_CURTOKLEN, TOKEN [PAR_SYMBOL])
191 1195 4 THEN
192 1196 5 RETURN (1)
193 1197 4 ELSE
194 1198 5 RETURN (0)
195 1199 5
196 1200 4 ELSE
197 1201 4 RETURN (0);
198 1202 4
199 1203 3 END;
200 1204 2 END;
201 1205 2
202 1206 2 [OUTRANGE] :
203 1207 3 BEGIN
204 1208 3 ASSERT (0);
```


EDT\$PRISTOK
V04-000

EDT\$PRISTOK - test for particular token
EDT\$\$PA_TSTTOK - test for particular token

L 16
16-Sep-1984 01:18:30
14-Sep-1984 12:24:09

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]PRISTOK.BLI;1 Page 5
(3)

```
: 205      1209  3      RETURN (0);  
: 206      1210  2      END;  
: 207      1211  2      TES;  
: 208      1212  2  
: 209      1213  1      END;
```

! of routine EDT\$\$PA_TSTTOK

.TITLE EDT\$PRISTOK EDT\$PRISTOK - test for particular t
oken

.IDENT \V04-000\

.PSECT _EDT\$CODE,NOWRT, SHR, PIC,2

```
45 47 4E 41 48 61 00000 P.AAA: .BYTE 97  
42 00001 .ASCII \CHANGE\  
43 00007 .BYTE 66  
59 50 4F 43 00008 .ASCII \COPY\  
63 0000C .BYTE 99  
45 4E 49 46 45 44 0000D .ASCII \DEFINE\  
61 00013 .BYTE 97  
45 54 45 4C 45 44 00014 .ASCII \DELETE\  
42 0001A .BYTE 66  
54 49 58 45 0001B .ASCII \EXIT\  
41 0001F .BYTE 65  
44 4E 49 46 00020 .ASCII \FIND\  
38 00024 .BYTE 59  
52 4F 46 00025 .ASCII \FOR\  
41 00028 .BYTE 65  
50 4C 45 48 00029 .ASCII \HELP\  
73 0002D .BYTE 115  
45 44 55 4C 43 4E 49 0002E .ASCII \INCLUDE\  
61 00035 .BYTE 97  
54 52 45 53 4E 49 00036 .ASCII \INSERT\  
31 0003C .BYTE 49  
59 45 4B 0003D .ASCII \KEY\  
61 00040 .BYTE 97  
44 41 50 59 45 4B 00041 .ASCII \KEYPAD\  
83 00047 .BYTE -125  
44 41 50 59 45 4B 4F 4E 00048 .ASCII \NOKEYPAD\  
41 00050 .BYTE 65  
45 56 4F 4D 00051 .ASCII \MOVE\  
51 00055 .BYTE 81  
54 4E 49 52 50 00056 .ASCII \PRINT\  
44 0005B .BYTE 68  
54 49 55 51 0005C .ASCII \QUIT\  
71 00060 .BYTE 113  
45 43 41 4C 50 45 52 00061 .ASCII \REPLACE\  
A3 00068 .BYTE -93  
45 43 4E 45 55 51 45 53 45 52 00069 .ASCII \RESEQUENCE\  
32 00073 .BYTE 50  
54 45 53 00074 .ASCII \SET\  
42 00077 .BYTE 66  
57 4F 48 53 00078 .ASCII \SHOW\  
A1 0007C .BYTE -95  
45 54 55 54 49 54 53 42 55 53 0007D .ASCII \SUBSTITUTE\  
41 00087 .BYTE 65  
45 50 59 54 00088 .ASCII \TYPE\  
:
```

				52	0008C	.BYTE	82	:
	45	54	49	52	57	.ASCII	\WRITE\	:
				61	00092	.BYTE	97	:
59	46	49	52	45	56	.ASCII	\VERIFY\	:
				11	00099	.BYTE	17	:
				23	0009A	.ASCII	\#\	:
				2A	0009B	.BYTE	42	:
			4F	54	0009C	.ASCII	\TO\	:
				4C	0009E	.BYTE	76	:
	55	52	48	54	0009F	.ASCII	\THRU\	:
				11	000A3	.BYTE	17	:
				2D	000A4	.ASCII	\-\	:
				11	000A5	.BYTE	17	:
				2C	000A6	.ASCII	\ \	:
				41	000A7	.BYTE	65	:
	54	58	45	4E	000A8	.ASCII	\NEXT\	:
				3B	000AC	.BYTE	59	:
			44	4E	41	.ASCII	\AND\	:
				11	000B0	.BYTE	17	:
				2B	000B1	.ASCII	\+\	:
				11	000B2	.BYTE	17	:
				3D	000B3	.ASCII	\=\	:
				11	000B4	.BYTE	17	:
				2E	000B5	.ASCII	\.\	:
				6A	000B6	.BYTE	106	:
52	45	46	46	55	42	.ASCII	\BUFFER\	:
				39	000BD	.BYTE	57	:
			44	4E	45	.ASCII	\END\	:
				59	000C1	.BYTE	89	:
4E	49	47	45	42	000C2	.ASCII	\BEGIN\	:
				49	000C7	.BYTE	73	:
	54	53	41	4C	000C8	.ASCII	\LAST\	:
				01	000CC	.BYTE	1	:
					000CD	.BLKB	0	:
				39	000CD	.BYTE	57	:
			4C	4C	41	.ASCII	\ALL\	:
				11	000D1	.BYTE	17	:
				3A	000D2	.ASCII	\:\	:
				49	000D3	.BYTE	73	:
		54	53	45	52	.ASCII	\REST\	:
				6B	000D8	.BYTE	107	:
45	52	4F	46	45	42	.ASCII	\BEFORE\	:
				59	000DF	.BYTE	89	:
	45	4C	4F	48	57	.ASCII	\WHOLE\	:
				01	000E5	.BYTE	1	:
					000E6	.BLKB	0	:
				00	000E6	.BYTE	0	:
					000E7	.BLKB	0	:
				11	000E7	.BYTE	17	:
				2F	000E8	.ASCII	\/\	:
				51	000E9	.BYTE	81	:
	59	52	45	55	51	.ASCII	\QUERY\	:
				51	000EF	.BYTE	81	:
	46	45	49	52	42	.ASCII	\BRIEF\	:
				83	000F5	.BYTE	-125	:
45	43	4E	45	55	51	.ASCII	\SEQUENCE\	:
				91	000FE	.BYTE	-111	:

45	54	41	43	49	4C	50	55	44	000FF	.ASCII	\DUPLICATE\	:
								63	00108	.BYTE	99	:
			45	50	59	54	4F	4E	00109	.ASCII	\NOTYPE\	:
								42	0010F	.BYTE	66	:
					45	56	41	53	00110	.ASCII	\SAVE\	:
								42	00114	.BYTE	66	:
					59	41	54	53	00115	.ASCII	\STAY\	:
								72	00119	.BYTE	114	:
		53	52	45	42	4D	55	4E	0011A	.ASCII	\NUMBERS\	:
								93	00121	.BYTE	-109	:
53	52	45	42	4D	55	4E	4F	4E	00122	.ASCII	\NONNUMBERS\	:
								42	0012B	.BYTE	66	:
					45	53	41	43	0012C	.ASCII	\CASE\	:
								52	00130	.BYTE	82	:
					54	43	41	58	00131	.ASCII	\EXACT\	:
								62	00136	.BYTE	98	:
				48	43	52	41	45	00137	.ASCII	\SEARCH\	:
								71	0013D	.BYTE	113	:
		4C	41	52	45	4E	45	47	0013E	.ASCII	\GENERAL\	:
								72	00145	.BYTE	114	:
		44	45	44	4E	55	4F	42	00146	.ASCII	\BOUNDED\	:
								91	0014D	.BYTE	-111	:
44	45	44	4E	55	4F	42	4E	55	0014E	.ASCII	\UNBOUNDED\	:
								82	00157	.BYTE	-126	:
	4C	41	4E	49	4D	52	45	54	00158	.ASCII	\TERMINAL\	:
								51	00160	.BYTE	81	:
				52	45	50	50	55	00161	.ASCII	\UPPER\	:
								51	00166	.BYTE	81	:
				52	45	57	4F	4C	00167	.ASCII	\LOWER\	:
								41	0016C	.BYTE	65	:
					45	4E	4F	4E	0016D	.ASCII	\NONE\	:
								44	00171	.BYTE	68	:
					32	35	54	56	00172	.ASCII	\VT52\	:
								55	00176	.BYTE	85	:
				30	30	31	54	56	00177	.ASCII	\VT100\	:
								44	0017C	.BYTE	68	:
					59	50	43	48	0017D	.ASCII	\HCPY\	:
								61	00181	.BYTE	97	:
			52	4F	53	52	55	43	00182	.ASCII	\CURSOR\	:
								41	00188	.BYTE	65	:
					44	4C	4F	47	00189	.ASCII	\GOLD\	:
								73	0018D	.BYTE	115	:
		4C	4F	52	54	4E	4F	43	0018E	.ASCII	\CONTROL\	:
								22	00195	.BYTE	34	:
							53	41	00196	.ASCII	\AS\	:
								83	00198	.BYTE	-125	:
		59	46	49	52	45	56	4F	00199	.ASCII	\NOVERIFY\	:
								82	001A1	.BYTE	-126	:
		45	54	41	43	4E	55	52	001A2	.ASCII	\TRUNCATE\	:
								A4	001AA	.BYTE	-92	:
45	54	41	43	4E	55	52	54	4F	001AB	.ASCII	\NOTRUNCATE\	:
								72	001B5	.BYTE	114	:
			4E	4F	49	53	52	45	001B6	.ASCII	\VERSION\	:
								41	001BD	.BYTE	65	:
					50	41	52	57	001BE	.ASCII	\WRAP\	:
								41	001C2	.BYTE	65	:
					44	52	4F	57	001C3	.ASCII	\WORD\	:

					63	001C7	.BYTE	99	:
	50	41	52	57	4F	4E 001C8	.ASCII	\NOWRAP\	:
					62	001CE	.BYTE	98	:
	4E	45	45	52	43	53 001CF	.ASCII	\SCREEN\	:
					51	001D5	.BYTE	81	:
		53	45	4E	49	4C 001D6	.ASCII	\LINES\	:
					41	001DB	.BYTE	65	:
			45	44	4F	4D 001DC	.ASCII	\MODE\	:
					51	001E0	.BYTE	81	:
		4F	52	43	41	4D 001E1	.ASCII	\MACRO\	:
					62	001E6	.BYTE	98	:
		59	54	49	54	4E 45 001E7	.ASCII	\ENTITY\	:
					43	001ED	.BYTE	67	:
			45	47	41	50 001EE	.ASCII	\PAGE\	:
					81	001F2	.BYTE	-127	:
	45	43	4E	45	54	4E 45 53 001F3	.ASCII	\SENTENCE\	:
					93	001FB	.BYTE	-109	:
48	50	41	52	47	41	52 41 50 001FC	.ASCII	\PARAGRAPH\	:
					52	00205	.BYTE	82	:
			52	41	45	4C 43 00206	.ASCII	\CLEAR\	:
					81	0020B	.BYTE	-127	:
	4C	41	4E	49	47	49 52 4F 0020C	.ASCII	\ORIGINAL\	:
					51	00214	.BYTE	81	:
			54	45	49	55 51 00215	.ASCII	\QUIET\	:
					73	0021A	.BYTE	115	:
		54	45	49	55	51 4F 4E 0021B	.ASCII	\NOQUIET\	:
					61	00222	.BYTE	97	:
		54	43	45	4C	45 53 00223	.ASCII	\SELECT\	:
					32	00229	.BYTE	50	:
				42	41	54 0022A	.ASCII	\TAB\	:
					53	0022D	.BYTE	83	:
			42	41	54	4F 4E 0022E	.ASCII	\NOTAB\	:
					43	00233	.BYTE	67	:
			4C	4C	49	46 00234	.ASCII	\FILL\	:
					61	00238	.BYTE	97	:
		54	53	55	4A	44 41 00239	.ASCII	\ADJUST\	:
					63	0023F	.BYTE	99	:
		45	54	45	4C	45 44 00240	.ASCII	\DELETE\	:
					63	00246	.BYTE	99	:
		54	41	45	50	45 52 00247	.ASCII	\REPEAT\	:
					85	0024D	.BYTE	-123	:
	54	41	45	50	45	52 4F 4E 0024E	.ASCII	\NOREPEAT\	:
					33	00256	.BYTE	51	:
				46	4E	46 00257	.ASCII	\FNF\	:
					55	0025A	.BYTE	85	:
			46	4E	46	4F 4E 0025B	.ASCII	\NOFNF\	:
					33	00260	.BYTE	51	:
				53	50	57 00261	.ASCII	\WPS\	:
					72	00264	.BYTE	114	:
		59	52	41	4D	4D 55 53 00265	.ASCII	\SUMMARY\	:
					94	0026C	.BYTE	-108	:
59	52	41	4D	4D	55	53 4F 4E 0026D	.ASCII	\NOSUMMARY\	:
					61	00276	.BYTE	97	:
		54	50	4D	4F	52 50 00277	.ASCII	\PROMPT\	:
					44	0027D	.BYTE	68	:
			45	4E	49	4C 0027E	.ASCII	\LINE\	:
					66	00282	.BYTE	102	:

				44	41	50	59	45	4B	00283	.ASCII	\KEYPAD\	:			
									87	00289	.BYTE	-121	:			
		44	41	50	59	45	4B	4F	4E	0028A	.ASCII	\NOKEYPAD\	:			
									87	00292	.BYTE	-121	:			
		45	47	4E	41	48	43	43	48	00293	.ASCII	\HCCHANGE\	:			
									66	0029B	.BYTE	102	:			
				54	52	45	53	4E	49	0029C	.ASCII	\INSERT\	:			
									77	002A2	.BYTE	119	:			
				4E	54	52	45	53	4E	49	002A3	.ASCII	\INSERTN\	:		
									55	002AA	.BYTE	85	:			
						59	52	45	55	51	002AB	.ASCII	\QUERY\	:		
									43	002B0	.BYTE	67	:			
							54	58	45	54	002B1	.ASCII	\TEXT\	:		
									33	002B5	.BYTE	51	:			
								44	4E	45	002B6	.ASCII	\END\	:		
									42	002B9	.BYTE	66	:			
						45	47	41	50	002BA	.ASCII	\PAGE\	:			
									42	002BE	.BYTE	66	:			
						44	52	4F	57	002BF	.ASCII	\WORD\	:			
									92	002C3	.BYTE	-110	:			
		52	45	54	49	4D	49	4C	45	44	002C4	.ASCII	\DELIMITER\	:		
									84	002CD	.BYTE	-76	:			
52	45	54	49	4D	49	4C	45	44	4F	4E	002CE	.ASCII	\NODELIMITER\	:		
									92	002D9	.BYTE	-110	:			
		48	50	41	52	47	41	52	41	50	002DA	.ASCII	\PARAGRAPH\	:		
									32	002E3	.BYTE	50	:			
							53	50	57	54	002E4	.ASCII	\WPS\	:		
									4E	002E7	.BYTE	84	:			
						53	50	57	4F	4E	002E8	.ASCII	\NOWPS\	:		
									A7	002ED	.BYTE	-89	:			
		54	41	45	50	45	52	4F	54	55	41	002EE	.ASCII	\AUTOREPEAT\	:	
									C7	002F8	.BYTE	-57	:			
54	41	45	50	45	52	4F	54	55	41	4F	4E	002F9	.ASCII	\NOAUTOREPEAT\	:	
									77	00305	.BYTE	119	:			
						44	4E	41	4D	4D	4F	43	00306	.ASCII	\COMMAND\	:
									52	0030D	.BYTE	82	:			
						53	45	4C	49	46	0030E	.ASCII	\FILES\	:		
									B7	00313	.BYTE	-73	:			
		4C	41	43	49	54	49	52	43	41	49	44	00314	.ASCII	\DIACRITICAL\	:
									B7	0031F	.BYTE	-73	:			
		45	56	49	54	49	53	4E	45	53	4E	49	00320	.ASCII	\INSENSITIVE\	:
									22	0032B	.BYTE	34	:			
									49	43	0032C	.ASCII	\CI\	:		
									22	0032E	.BYTE	34	:			
									49	44	0032F	.ASCII	\DI\	:		
									83	00331	.BYTE	-125	:			
				4E	4F	49	54	43	4E	55	46	00332	.ASCII	\FUNCTION\	:	
									66	0033A	.BYTE	102	:			
						4C	4C	4F	52	43	53	0033B	.ASCII	\SCROLL\	:	
									87	00341	.BYTE	-121	:			
				4C	4C	4F	52	43	53	4F	4E	00342	.ASCII	\NOSCROLL\	:	
									87	0034A	.BYTE	-121	:			
				54	49	42	54	48	47	49	45	0034B	.ASCII	\EIGHTBIT\	:	
									A7	00353	.BYTE	-89	:			
		54	49	42	54	48	47	49	45	4F	4E	00354	.ASCII	\NOEIGHTBIT\	:	
									44	0035E	.BYTE	68	:			
						54	49	44	45	0035F	.ASCII	\EDIT\	:			

										54	49	44	45	4F	66	00363
															4E	00364
															00	0036A
																0036B
01C2	01CE	01BD	0028	0229	00CC	01D5	012B	00D1	00E5							0036C
0256	01ED	01F2	01DB	01E6	0246	023F	01FB	0214	01A1							00380
0181	0136	0157	003C	00A5	0040	0092	00B6	00AC	00A7							00394
0289	0282	02A2	029B	0292	0000	0119	00BD	00C1	018D							003A8
0305	0276	0331	02B5	02AA	02B0	02D9	02BE	02B9	02ED							003BC
00D3	00A3	009E	009B	0099	0341	034A	033A	027D	0264							003D0
030D	00C7	00CD	00B0	00B4	035E	0353	0363	00B2	00D8							003E4
0024	000C	0013	001A	032E	0035	032B	031F	002D	00DF							003F8
0050	0055	008C	007C	0087	0068	0073	0077	0007	001F							0040C
0188	01E0	01B5	01C7	0198	01AA	0313	005B	0060	0047							00420
0260	026C	024D	0233	0238	0176	02C3	0195	017C	02CD							00434
010F	0114	00FE	0108	025A	0222	022D	0205	020B	021A							00448
016C	014D	0160	02F8	00EF	00F5	00E6	00E7	00E9	0121							0045C
			013D	0145	02E7	0130	02E3	0171	0166							00470

```

      .BYTE
      .ASCII
      .BYTE
      .BLKB
P.AAB: .WORD

```

```

102
\NOEDIT\
0
1
229, 209, 299, 469, 204, 553, 40, 445, -
462, 450, 417, 532, 507, 575, 582, 486, -
475, 498, 493, 598, 167, 172, 182, 146, -
64, 165, 60, 343, 310, 385, 397, 193, -
189, 281, 0, 658, 667, 674, 642, 649, -
749, 697, 702, 729, 688, 682, 693, 817, -
630, 773, 612, 637, 826, 842, 833, 153, -
155, 158, 163, 211, 216, 178, 867, 851, -
862, 180, 176, 205, 199, 781, 223, 45, -
799, 811, 53, 814, 26, 19, 12, 36, 31, 7,
119, 115, 104, 135, 124, 140, 85, 80, 71,
96, 91, 787, 426, 408, 455, 437, 480, -
392, 717, 380, 405, 707, 374, 568, 563, -
589, 620, 608, 538, 523, 517, 557, 546, -
602, 264, 254, 276, 271, 289, 233, 231, -
230, 245, 239, 760, 352, 333, 364, 358, -
369, 739, 304, 743, 325, 317

```

TOKEN_TABLE=
TOK_INDX=

P.AAA
P.AAB

```

      .EXTRN  EDT$PA_CURTOK
      .EXTRN  EDT$G_PA_CURTOKLEN
      .EXTRN  EDT$G_PA_PCEN
      .EXTRN  EDT$G_PA_TOKCLASS
      .EXTRN  EDT$INTER_ERR

```

0017 03 00 00000000G 00 000c 00000
001D 0011 001D CF 00002
0000A 1\$:

```
.ENTRY EDT$PA TSTOK, Save R2,R3
CASEL EDT$G_PA_TOKCLASS, #0, #3
.WORD 4$-1$, =
```

0017

03
001D

00
0011

00000000G

00
001D

CF 00002
0000A

18:

00000000G	00	00	FB	00012	
		4E	11	00019	
01	04	AC	D1	0001B	2\$:
		42	11	0001F	
05	04	AC	D1	00021	3\$:
		3C	11	00025	
50	04	AC	D0	00027	4\$:
50	FEBC	CF40	3C	0002B	
52	FB4C	CF40	9E	00031	
04	00000000G	00	E9	00037	
62		03	E1	0003E	
50	00000000G	00	D0	00042	5\$:
03		00	ED	00049	
		19	14	0004E	
04		04	ED	00050	
		12	19	00055	
51	00000000G	00	D0	00057	
61		50	29	0005E	
		04	12	00063	6\$:
50		01	D0	00065	
			04	00068	

```

CALLS      #0, EDT$$INTER_ERR
BRB        7$
CMPL       TOK_NUM, #1
BRB        6$
CMPL       TOK_NUM, #5
BRB        6$
MOVL       TOK_NUM, R0
MOVZWL     TOK_INDX-2[R0], TOK
MOVAB      TOKEN_TABLE[TOK], R2
BLBC       EDT$$G_PA_PCEN, 5$
BBC        #3, (R2), -7$
MOVL       EDT$$G_PA_CURTOKLEN, R0
CMPZV      #0, #3, (R2), R0
BGTR       7$
CMPZV      #4, #4, (R2), R0
BLSS       7$
MOVL       EDT$$A_PA_CURTOK, R1
CMPC3      R0, (R1), -1(R2)
BNEQ       7$
MOVL       #1, R0
RET

```

50
50

01 A2

EDT\$PRISTOK
VO4-000

EDT\$PRISTOK - test for particular token
EDT\$\$PA_TSTTOK - test for particular token

F 1
16-Sep-1984 01:18:30
14-Sep-1984 12:24:09

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]PRISTOK.BLI;1 Page 11 (3)

50 D4 00069 7\$: CLRL R0
04 0006B RET

; 1213
;

; Routine Size: 108 bytes, Routine Base: _EDT\$CODE + 047E

; 210 1214 1
; 211 1215 1 !<BLF/PAGE>

EDT\$PRISTOK
V04-000

EDT\$PRISTOK - test for particular token
EDT\$SPA_TSTTOK - test for particular token

G 1
16-Sep-1984 01:18:30
14-Sep-1984 12:24:09

VAX-11 Bliss-32 V4.0-742
DISK\$VMMASTER:[EDT.SRC]PRISTOK.BLI;1
Page 12
(4)

: 213
: 214
: 215
1216 1 END
1217 1
1218 0 ELUDOM

! of module EDT\$PRISTOK

PSECT SUMMARY

Name Bytes Attributes
_EDT\$CODE 1258 NOVEC,NOWRT. RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[EDT.SRC]EDT.L32;1	377	9	2	40	00:00.2
_\$255\$DJA28:[EDT.SRC]PSECTS.L32;1	2	1	50	7	00:00.1

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS\$:PRISTOK/OBJ=OBJ\$:PRISTOK MSRC\$:PRISTOK.BLI/UPDATE=(ENH\$:PRISTOK)

: Size: 108 code + 1150 data bytes
: Run Time: 00:23.2
: Elapsed Time: 00:29.6
: Lines/CPU Min: 3154
: Lexemes/CPU-Min: 11132
: Memory Used: 121 pages
: Compilation Complete

0137

AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

0138

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY