

[illegible]

EXHIBIT

SSSSSSSS	CCCCCCCC	RRRRRRRR	FFFFFFFF	IIIIII	NN	NN	DDDDDDDD	
SSSSSSSS	CCCCCCCC	RRRRRRRR	FFFFFFFF	IIIIII	NN	NN	DDDDDDDD	
SS	CC	RR	FF	II	NN	NN	DD	DD
SS	CC	RR	FF	II	NN	NN	DD	DD
SS	CC	RR	FF	II	NNNN	NN	DD	DD
SS	CC	RR	FF	II	NNNN	NN	DD	DD
SSSSSS	CC	RRRRRRRR	FFFFFF	II	NN	NN	DD	DD
SSSSSS	CC	RRRRRRRR	FFFFFF	II	NN	NN	DD	DD
SS	CC	RR	FF	II	NN	NN	DD	DD
SS	CC	RR	FF	II	NN	NNNN	DD	DD
SS	CC	RR	FF	II	NN	NNNN	DD	DD
SS	CC	RR	FF	II	NN	NN	DD	DD
SSSSSSSS	CCCCCCCC	RR	FF	IIIIII	NN	NN	DDDDDDDD	
SSSSSSSS	CCCCCCCC	RR	FF	IIIIII	NN	NN	DDDDDDDD	
								
								

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
LLLLLLLLLL	IIIIII	SSSSSSSS
LLLLLLLLLL	IIIIII	SSSSSSSS

```
0001 0 %TITLE 'EDT$SCRFIND - find a record in the screen structure'
0002 0 MODULE EDT$SCRFIND ( ! Find a screen record
0003 0 IDENT = 'V04-000' ! File: SCRFIND.BLI Edit: JBS1016
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 This module searches for a record in the screen
0037 1 data structure.
0038 1
0039 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0040 1
0041 1 AUTHOR: John Sauter CREATION DATE: 13-Sep-1982
0042 1
0043 1 MODIFIED BY:
0044 1
0045 1 1-001 - Original. This module was created by extracting the search code from
0046 1 EDT$CHMMRKCHG. SMB 22-Sep-1982
0047 1 1-002 - Add some parameters and extra code to original extracted version.
0048 1 SMB 22-Sep-1982
0049 1 1-003 - Change the parameters passed. SMB 28-Sep-1982
0050 1 1-004 - Modify to use new 48 bit macros. SIS 01-Oct-1982
0051 1 1-005 - Search for EOB differently. SMB 05-Oct-1982
0052 1 1-006 - Modify to return value of screen pointer found. SMB 08-Oct-1982
0053 1 1-007 - Increment and decrement displacement on lines equal. SMB 12-Oct-1982
0054 1 1-008 - Modify to use new compare macro. SIS 20-Oct-1982
0055 1 1-009 - Deleted lines have the record number of the next line. JBS 25-Oct-1982
0056 1 1-010 - Modify search for bottom line of a record. SMB 28-Oct-1982
0057 1 1-011 - Fix a bug in nottruncate mode. SMB 24-Nov-1982
```

EDT\$SCRFIND
V04-000

J 16
EDT\$SCRFIND - find a record in the screen struc 16-Sep-1984 01:33:52 VAX-11 Bliss-32 V4.0-742 Page 2
14-Sep-1984 12:24:26 DISK\$VMSMASTER:[EDT.SRC]SCRFIND.BLI;1 (1)

:	58	0058	1	:	1-012	-	Worry about deleted lines when scanning backwards.	JBS 25-Nov-1982
:	59	0059	1	:	1-013	-	Remove the edit buffer.	JBS 27-Dec-1982
:	60	0060	1	:	1-014	-	Fix a problem involving deleted lines in nottruncate mode.	JBS 23-May-1983
:	61	0061	1	:	1-015	-	Improve the appearance of the listing.	JBS 17-Jun-1983
:	62	0062	1	:	1-016	-	The delete flag takes priority over column position.	JBS 06-Jul-1983
:	63	0063	1	:	--			
:	64	0064	1	:				

EDT\$SCRFIND
V04-000

EDT\$SCRFIND - find a record in the screen struc
Declarations

K 16
16-Sep-1984 01:33:52
14-Sep-1984 12:24:26

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]SCRFIND.BLI;1 Page 3
(2)

```

: 66      0065 1 %SBTTL 'Declarations'
: 67      0066 1
: 68      0067 1 : TABLE OF CONTENTS:
: 69      0068 1 :
: 70      0069 1
: 71      0070 1 REQUIRE 'EDTSRC:TRAROUNAM';
: 72      0509 1
: 73      0510 1 FORWARD ROUTINE
: 74      0511 1     EDT$SC_FNDREC;
: 75      0512 1
: 76      0513 1 :
: 77      0514 1 : INCLUDE FILES:
: 78      0515 1 :
: 79      0516 1
: 80      0517 1 REQUIRE 'EDTSRC:EDTREQ';
: 81      0652 1
: 82      0653 1 :
: 83      0654 1 : MACROS:
: 84      0655 1 :
: 85      0656 1 :     NONE
: 86      0657 1 :
: 87      0658 1 : EQUATED SYMBOLS:
: 88      0659 1 :
: 89      0660 1 :     NONE
: 90      0661 1 :
: 91      0662 1 : OWN STORAGE:
: 92      0663 1 :
: 93      0664 1 :     NONE
: 94      0665 1 :
: 95      0666 1 : EXTERNAL REFERENCES:
: 96      0667 1 :
: 97      0668 1 :     In the routine
```

! Fins a screen record

EDTSSCRFIND
V04-000

L 16

EDTSSCRFIND - find a record in the screen struc 16-Sep-1984 01:33:52 VAX-11 Bliss-32 14.0-742 Page 4
EDTSSSC_FNDREC - find a record in the screen da 14-Sep-1984 12:24:26 DISK\$VMSMASTER:[EDT.SRC]SCRFINDBLI;1 (3)

```

99 0669 1 %SBTTL 'EDTSSSC_FNDREC - find a record in the screen data structure'
100 0670 1
101 0671 1 GLOBAL ROUTINE EDTSSSC_FNDREC (           ! Find a screen record
102 0672 1     POSITION,                          ! Position on the line
103 0673 1     DISPLACEMENT                      ! Displacement from cursor screen line
104 0674 1 ) : =
105 0675 1
106 0676 1 ++
107 0677 1 FUNCTIONAL DESCRIPTION:
108 0678 1
109 0679 1     Given the current record number, determine if it has a corresponding
110 0680 1     screen block in the screen data structure.  If so, return
111 0681 1     the address.
112 0682 1
113 0683 1 FORMAL PARAMETERS:
114 0684 1
115 0685 1     POSITION                The position on the line at which the operation takes place
116 0686 1
117 0687 1     DISPLACEMENT            The displacement of the current line info block from the cursor line info block
118 0688 1
119 0689 1 IMPLICIT INPUTS:
120 0690 1
121 0691 1     EDTSSA_LST_SCRPTR
122 0692 1     EDTSSA_EOB_SCRPTR
123 0693 1     EDTSSA_WK_LN
124 0694 1     EDTSSZ_EOB_LN
125 0695 1     EDTSSA_CSR_SCRPTR
126 0696 1     EDTSSA_CUR_BUF
127 0697 1     EDTSSL_CUR_SCRLN
128 0698 1     EDTSSA_SCR_BUF
129 0699 1
130 0700 1 IMPLICIT OUTPUTS:
131 0701 1
132 0702 1     NONE
133 0703 1
134 0704 1 ROUTINE VALUE:
135 0705 1
136 0706 1     NONE
137 0707 1
138 0708 1 SIDE EFFECTS:
139 0709 1
140 0710 1     NONE
141 0711 1
142 0712 1 --
143 0713 1
144 0714 2 BEGIN
145 0715 2
146 0716 2 EXTERNAL
147 0717 2     EDTSSA_LST_SCRPTR : REF SCREEN_LINE, ! Last screen info pointer
148 0718 2     EDTSSZ_EOB_LN, ! Address of EOB line
149 0719 2     EDTSSA_WK_LN, ! Current workfile line
150 0720 2     EDTSSA_EOB_SCRPTR : REF SCREEN_LINE, ! EOB screen pointer
151 0721 2     EDTSSA_CSR_SCRPTR : REF SCREEN_LINE, ! Pointer to cursor line screen info
152 0722 2     EDTSSA_CUR_BUF : REF TBCB_BLOCK, ! Current text buffer control block
153 0723 2     EDTSSL_CUR_SCRLN : LN_BLOCK, ! Record number of current screen line
154 0724 2     EDTSSA_SCR_BUF : REF TBCB_BLOCK; ! Current screen buffer
155 0725 2
```


EDTSSCRFIND
V04-000

EDTSSCRFIND - find a record in the screen struc 16-Sep-1984 01:33:52
EDTSSC_FNDREC - find a record in the screen da 14-Sep-1984 12:24:26

M 16

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]SCRFINDBLI;1

Page 5
(3)

```

156 0726 2 LOCAL
157 0727 2 DISP
158 0728 2 SCRPT : REF SCREEN_LINE, ! Displacement temporary
159 0729 2 RECORD_NO : LN_BLOCK, ! Pointer to our screen line
160 0730 2 FOUND_LINE; ! Record number of our screen line
161 0731 2
162 0732 2
163 0733 2 !+ If the screen buffer is different from the current buffer then bail out.
164 0734 2 ! None of the lines in the current buffer can be on the screen.
165 0735 2 !-
166 0736 2 DISP = 0;
167 0737 2 FOUND_LINE = 0;
168 0738 2
169 0739 3 IF (.EDTSSA_CUR_BUF EQLA .EDTSSA_SCR_BUF)
170 0740 2 THEN
171 0741 3 BEGIN
172 0742 3 !+ Find our screen line block.
173 0743 3 !-
174 0744 3 SCRPT = .EDTSSA_CSR_SCRPT;
175 0745 3
176 0746 3 !+ First check to see if we are searching for EOB on the screen.
177 0747 3 !-
178 0748 3
179 0749 3 IF (.EDTSSA_WK_LN EQLA EDTSSZ_EOB_LN)
180 0750 4 THEN
181 0751 3 BEGIN
182 0752 4
183 0753 4 IF (.EDTSSA_LST_SCRPT NEQA .EDTSSA_EOB_SCRPT)
184 0754 5 THEN
185 0755 4 FOUND_LINE = 0
186 0756 4 ELSE
187 0757 4 BEGIN
188 0758 5 FOUND_LINE = 1;
189 0759 5
190 0760 5 WHILE (.SCRPT NEQA .EDTSSA_EOB_SCRPT) DO
191 0761 5 BEGIN
192 0762 6 SCRPT = .SCRPT [SCR_NXT_LINE];
193 0763 6 DISP = .DISP + 1;
194 0764 6 END;
195 0765 5
196 0766 5 END;
197 0767 4
198 0768 4 END
199 0769 4 ELSE
200 0770 3 BEGIN
201 0771 4 MOVE LINE (EDTSSL_CUR_SCRLN, RECORD_NO);
202 0772 4
203 0773 4 WHILE (( NOT .FOUND_LINE) AND (.SCRPT NEQA 0)) DO
204 0774 4 CASE CMLNO (RECORD_NO, EDTSSA_CUR_BUF [TBCB_CUR_LIN]) FROM -1 TO 1 OF
205 0775 4 SET
206 0776 4 [-1] : ! Cursor line is before current line
207 0777 4 BEGIN
208 0778 4
209 0779 4
210 0780 5
211 0781 5
212 0782 5 LOCAL
```

```
213 0783 5      NEXT_SCRPTR : REF SCREEN_LINE;
214 0784 5
215 0785 5      NEXT_SCRPTR = .SCRPTR [SCR_NXT_LINE];
216 0786 5
217 0787 6      IF (.NEXT_SCRPTR NEQA 0)
218 0788 5      THEN
219 0789 6          BEGIN
220 0790 6
221 0791 7          IF (((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0) AND
222 0792 8              ((.NEXT_SCRPTR[SCR_LINE_IDX] EQL 0) OR
223 0793 7              ((.NEXT_SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0)))
224 0794 6          THEN
225 0795 6              ADDLINE (NUMBER_ONE, RECORD_NO);
226 0796 6
227 0797 5          END;
228 0798 5
229 0799 5      DISP = .DISP + 1;
230 0800 5      SCRPTR = .SCRPTR [SCR_NXT_LINE];
231 0801 4      END;
232 0802 4
233 0803 4      [0] :          ! Cursor line is the current line
234 0804 5      BEGIN
235 0805 5      !+
236 0806 5      ! If this line is deleted test the next following non-deleted line.
237 0807 5      !-
238 0808 5
239 0809 5      WHILE ((.SCRPTR NEQA 0) AND ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0)) D
240 0810 5          SCRPTR = .SCRPTR [SCR_NXT_LINE];
241 0811 5
242 0812 6      IF (.SCRPTR NEQA 0)
243 0813 5      THEN
244 0814 6          BEGIN
245 0815 6      !+
246 0816 6      ! We have a non-deleted line, check column number.
247 0817 6      !-
248 0818 6
249 0819 8          IF ((.SCRPTR [SCR_CHR_FROM] LEQ .POSITION)
250 0820 7              AND (.SCRPTR [SCR_CHR_TO] GEQ .POSITION))
251 0821 6          THEN
252 0822 6      !+
253 0823 6      ! This line is not deleted and contains the correct column. It is the line we want.
254 0824 6      !-
255 0825 6          FOUND_LINE = 1
256 0826 6      ELSE
257 0827 7          BEGIN
258 0828 7
259 0829 8          IF (.POSITION GTR .SCRPTR [SCR_CHR_TO])
260 0830 7          THEN
261 0831 8              BEGIN
262 0832 8      !+
263 0833 8      ! This line is continued; the line we want is further down the screen.
264 0834 8      !-
265 0835 8              SCRPTR = .SCRPTR [SCR_NXT_LINE];      ! Move forward
266 0836 8              DISP = .DISP + 1;
267 0837 8              END
268 0838 7          ELSE
269 0839 8              BEGIN
```



```
270 0840 8 1+
271 0841 8 1- This is a continuation line; the line we want is further up the screen.
272 0842 8 1-
273 0843 8
274 0844 8 ASSERT (.POSITION LSS .SCRPTR [SCR_CHR_FROM]);
275 0845 8 SCRPTR = .SCRPTR [SCR_PRV_LINE]; ! Move backward
276 0846 8 ASSERT ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0);
277 0847 7 DISP = .DISP - 1;
278 0848 7 END;
279 0849 6 END;
280 0850 6
281 0851 5 END;
282 0852 5
283 0853 4 END;
284 0854 4
285 0855 4 [1] : ! Cursor line is after the current line
286 0856 5 BEGIN
287 0857 5
288 0858 6 IF ((.SCRPTR [SCR_LINE_IDX] EQL 0) OR !
289 0859 6 ((.SCRPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) NEQ 0))
290 0860 5 THEN
291 0861 6 BEGIN
292 0862 6
293 0863 6 LOCAL
294 0864 6 PREV_SCPTR : REF SCREEN_LINE;
295 0865 6
296 0866 6 PREV_SCPTR = .SCRPTR [SCR_PRV_LINE];
297 0867 6
298 0868 7 IF (.PREV_SCPTR NEQA 0)
299 0869 6 THEN
300 0870 7 BEGIN
301 0871 7
302 0872 8 IF ((.PREV_SCPTR [SCR_EDIT_FLAGS] AND SCR_EDIT_DELLN) EQL 0)
303 0873 7 THEN
304 0874 7 SUBLINE (NUMBER_ONE, RECORD_NO);
305 0875 7
306 0876 6 END;
307 0877 6
308 0878 5 END;
309 0879 5
310 0880 5 DISP = .DISP - 1;
311 0881 5 SCRPTR = .SCRPTR [SCR_PRV_LINE];
312 0882 4 END;
313 0883 4
314 0884 4 [OUTRANGE] :
315 0885 4 ASSERT (0);
316 0886 4 TES;
317 0887 4
318 0888 3 END;
319 0889 3
320 0890 2 END;
321 0891 2
322 0892 2 1+
323 0893 2 1- If FOUND_LINE is not set, the record does not appear on the screen.
324 0894 2 1- In that case return zero.
325 0895 2
326 0896 2 .DISPLACEMENT = .DISP;
```

: Rc

: 2
: 2

EDT\$SCRFIND
V04-000

: 327
: 328
: 329
: 330

EDT\$SCRFIND - find a record in the screen struc
EDT\$\$\$SC_FNDREC - find a record in the screen da

0897 2
0898 2
0899 2
0900 1

IF .FOUND_LINE THEN RETURN (.SCRPTR) ELSE RETURN (0);

END;

! of routine EDT\$\$\$SC_FNDREC

.TITLE EDT\$SCRFIND EDT\$SCRFIND - find a record in the
screen struc

.IDENT \V04-000\

.EXTRN EDT\$\$A_LST_SCRPTR
.EXTRN EDT\$\$Z_EOB_LN, EDT\$\$A_WK_LN
.EXTRN EDT\$\$A_EOB_SCRPTR
.EXTRN EDT\$\$A_CSR_SCRPTR
.EXTRN EDT\$\$A_CUR_BUF, EDT\$\$L_CUR_SCRLN
.EXTRN EDT\$\$A_SCR_BUF, EDT\$\$INTER_ERR

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

.ENTRY EDT\$\$\$SC_FNDREC, Save R2,R3,R4,R5,R6,R7,R8,- : 0671
R9,R10

MOVAB EDT\$\$A_CUR_BUF, R10
MOVAB EDT\$\$INTER_ERR, R9
SUBL2 #8, SP

CLRQ DISP : 0736
CMPL EDT\$\$A_CUR_BUF, EDT\$\$A_SCR_BUF : 0739

BNEQ 1\$:
MOVL EDT\$\$A_CSR_SCRPTR, SCRPTR : 0745
MOVAB EDT\$\$Z_EOB_LN, R0 : 0750

CMPL EDT\$\$A_WK_LN, R0
BNEQ 4\$:
MOVL EDT\$\$A_EOB_SCRPTR, R0 : 0754

CMPL EDT\$\$A_LST_SCRPTR, R0
BEQL 2\$:
CLRL FOUND_LINE : 0756

BRW 26\$:
MOVL #1, FOUND_LINE : 0759
CMPL SCRPTR, R0 : 0761

BEQL 1\$:
MOVL 4(SCRPTR), SCRPTR : 0763
INCL DISP : 0764

BRB 3\$: 0761
MOVC3 #6, EDT\$\$L_CUR_SCRLN, RECORD_NO : 0772
BLBS FOUND_LINE, 1\$: 0774

TSTL SCRPTR
BEQL 1\$:
ADDL3 #6, EDT\$\$A_CUR_BUF, R0 : 0776

CMPL HIGH_1, 4(R0)
BLSSU 6\$:
BNEQ 8\$:
CMPL LOW_1, (R0) :
BGEQU 7\$:
MNEGL #1, R0 :
BRB 9\$:
BNEQ 8\$:
CLRL R0 :
BRB 9\$:

ADDL3 #6, EDT\$\$A_CUR_BUF, R0 : 0776

CMPL HIGH_1, 4(R0)

BLSSU 6\$

BNEQ 8\$

CMPL LOW_1, (R0)

BGEQU 7\$

MNEGL #1, R0

BRB 9\$

BNEQ 8\$

CLRL R0

BRB 9\$

EDT\$
V04-

: 2
: 2
: 2

Si
Ru
El
Li
Me
Co

			50	01	D0	00086	8\$:	MCVL	#1, R0		
			8F	50	CF	00089	9\$:	CASEL	R0, #-1, #2		
			002F	000B		00091	10\$:	.WORD	12\$-10\$,-		
									16\$-10\$,-		
									22\$-10\$		
			69	00	FB	00097		CALLS	#0, EDT\$\$INTER_ERR		0885
				06	11	0009A	11\$:	BRB	5\$		0776
			50	04	A6	D0	0C09C	12\$:	MOVL	4(SCRPTR), NEXT_SCRPTR	0785
					16	13	000A0		BEQL	14\$	0787
					02	E0	000A2		BBS	#2, 13(SCRPTR), 14\$	0791
				08	A0	95	000A7		TSTB	8(NEXT_SCRPTR)	0792
					05	13	000AA		BEQL	13\$	
					02	E1	000AC		BBC	#2, 13(NEXT_SCRPTR), 14\$	0793
					6E	D6	000B1	13\$:	INCL	FIRST_LWORD	0795
					03	12	000B3		BNEQ	14\$	
				04	AE	B6	000B5		INCL	NEXT_WORD	
					57	D6	000B8	14\$:	INCL	DISP	0799
			56	04	A6	D0	000BA		MOVL	4(SCRPTR), SCRPTR	0800
					A2	11	000BE	15\$:	BRB	5\$	0776
					56	D5	000C0	16\$:	TSTL	SCRPTR	0809
					9E	13	000C2		BEQL	5\$	
					02	E1	000C4		BBC	#2, 13(SCRPTR), 17\$	
				04	A6	D0	000C9		MOVL	4(SCRPTR), SCRPTR	0810
					F1	11	000CD		BRB	16\$	
					91	13	000CF	17\$:	BEQL	5\$	0812
					00	ED	000D1		CMPZV	#0, #8, 9(SCRPTR), POSITION	0819
					0E	14	000D8		BGTR	18\$	
					00	ED	00CDA		CMPZV	#0, #8, 10(SCRPTR), POSITION	0820
					05	19	000E1		BLSS	18\$	
					01	D0	000E3		MOVL	#1, FOUND_LINE	0825
					52	11	000E6		BRB	25\$	
					00	ED	000E8	18\$:	CMPZV	#0, #8, 10(SCRPTR), POSITION	0829
					08	18	000EF		BGEQ	19\$	
				04	A6	D0	000F1		MOVL	4(SCRPTR), SCRPTR	0835
					57	D6	000F5		INCL	DISP	0836
					A1	11	000F7		BRB	11\$	0829
					00	ED	000F9	19\$:	CMPZV	#0, #8, 9(SCRPTR), POSITION	0843
					03	14	00100		BGTR	20\$	
					00	FB	00102		CALLS	#0, EDT\$\$INTER_ERR	
					66	D0	00105	20\$:	MOVL	(SCRPTR), SCRPTR	0844
					02	E1	00108		BBC	#2, 13(SCRPTR), 21\$	0845
					00	FB	0010D		CALLS	#0, EDT\$\$INTER_ERR	
					57	D7	00110	21\$:	DECL	DISP	0846
					86	11	00112		BRB	11\$	0812
				08	A6	95	00114	22\$:	TSTB	8(SCRPTR)	0858
					05	13	00117		BEQL	23\$	
					02	E1	00119		BBC	#2, 13(SCRPTR), 24\$	0859
					56	D0	0011E	23\$:	MOVL	(SCRPTR), PREV_SCRPTR	0866
					12	13	00121		BEQL	24\$	0868
					02	E0	00123		BBS	#2, 13(PREV_SCRPTR), 24\$	0872
					6E	D0	00128		MOVL	FIRST_WORD, -SAVE	0874
					6E	D7	0012B		DECL	FIRST_WORD	
					6E	D1	0012D		CMPL	FIRST_WORD, SAVE	
					03	1B	00130		BLEQU	24\$	
				04	AE	B7	00132		DECV	NEXT_WORD	
					57	D7	00135	24\$:	DECL	DISP	0880
					66	D0	00137		MOVL	(SCRPTR), SCRPTR	0881

EDT\$SCRFIND
V04-000

F 1
16-Sep-1984 01:33:52

EDT\$SCRFIND - find a record in the screen struc
EDT\$SC_FNDREC - find a record in the screen da

14-Sep-1984 12:24:26

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[EDT.SRC]SCRFIND.BLI;1

Page 10
(3)

08	BC	82	11	0013A	25\$:	BRB	15\$	:	0776
	02	57	D0	0013C	26\$:	MOVL	DISP, @DISPLACEMENT	:	0896
		58	E8	00140		BLBS	FOUND_LINE, 27\$	:	0898
		56	D4	00143		CLRL	R6	:	
	50	56	D0	00145	27\$:	MOVL	R6, R0	:	
		04	00148			RET		:	0900

; Routine Size: 329 bytes, Routine Base: _EDT\$CODE + 0000

:	331	0901	1
:	332	0902	1 !<BLF/PAGE>

EDT\$SCRFIND
V04-000

EDT\$SCRFIND - find a record in the screen struc
EDT\$\$SC_FNDREC - find a record in the screen da

G 1

16-Sep-1984 01:33:52

VAX-11 Bliss-32 V4.0-742

Page 11

(4)

! of module EDT\$SCRFIND

: 334
: 335
: 336

0903 1 END
0904 1
0905 0 ELUDOM

EDT!
V04-

:

PSECT SUMMARY

Name

Bytes

Attributes

: _EDT\$CODE

329 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 47 12 40 00:00.2

: _\$255\$DUA28:[EDT.SRC]PSECTS.L32;1

2 1 50 7 00:00.1

:

COMMAND QUALIFIERS

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

: Size: 329 code + 0 data bytes

: Run Time: 00:22.7

: Elapsed Time: 00:26.2

: Lines/CPU Min: 2392

: Lexemes/CPU-Min: 11450

: Memory Used: 174 pages

: Compilation Complete

0138

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

0139 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

