


```

1 0001 0 %TITLE 'EDT$FCOLINC - compute formatted character width'
2 0002 0 MODULE EDT$FCOLINC ( ! Compute formatted character width
3 0003 0 IDENT = 'V04-000' ! File: FCOLINC.BLI Edit: JBS1008
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1 *****
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
11 0011 1 * ALL RIGHTS RESERVED.
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
18 0018 1 * TRANSFERRED.
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
22 0022 1 * CORPORATION.
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
26 0026 1 *
27 0027 1 *
28 0028 1 *****
29 0029 1
30 0030 1
31 0031 1 ++
32 0032 1 FACILITY: EDT -- The DEC Standard Editor
33 0033 1
34 0034 1 ABSTRACT:
35 0035 1
36 0036 1 Compute formatted character width.
37 0037 1
38 0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
39 0039 1
40 0040 1 AUTHOR: Bob Kushlis, CREATION DATE: March 18, 1979
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original. DJS 19-FEB-1981. This module was created by
45 0045 1 extracting routine EDT$$FMT.CHWID from module FORMAT.
46 0046 1 1-002 - Regularize headers. JBS 05-Mar-1981
47 0047 1 1-003 - Change length of form feed string. STS 08-Oct-1981
48 0048 1 1-004 - Add support for DEC STD 169. JBS 11-Aug-1982
49 0049 1 1-005 - Remove ".L32". JBS 12-Aug-1982
50 0050 1 1-006 - Add conditional for VT220 support. JBS 10-Feb-1983
51 0051 1 1-007 - Deduce the width of unusual characters from their representation table. JBS 04-Mar-1983
52 0052 1 1-008 - Correct the computation for characters on an 8-bit terminal. JBS 07-Mar-1983
53 0053 1 --
54 0054 1

```

```
: 56      0055 1 %SBTTL 'Declarations'  
: 57      0056 1  
: 58      0057 1 | TABLE OF CONTENTS:  
: 59      0058 1 |  
: 60      0059 1  
: 61      0060 1 REQUIRE 'EDTSRC:TRAROUNAM';  
: 62      0499 1  
: 63      0500 1 FORWARD ROUTINE  
: 64      0501 1     EDT$$FMT_CHWID;  
: 65      0502 1  
: 66      0503 1 |  
: 67      0504 1 | INCLUDE FILES:  
: 68      0505 1 |  
: 69      0506 1  
: 70      0507 1 REQUIRE 'EDTSRC:EDTREQ';  
: 71      0642 1  
: 72      0643 1 LIBRARY 'EDTSRC:SUPPORTS';  
: 73      0644 1  
: 74      0645 1 LIBRARY 'EDTSRC:TRANSLATE';  
: 75      0646 1  
: 76      0647 1 |  
: 77      0648 1 | MACROS:  
: 78      0649 1 |  
: 79      0650 1 |     NONE  
: 80      0651 1 |  
: 81      0652 1 | EQUATED SYMBOLS:  
: 82      0653 1 |  
: 83      0654 1 |     NONE  
: 84      0655 1 |  
: 85      0656 1 | OWN STORAGE:  
: 86      0657 1 |  
: 87      0658 1 |     NONE  
: 88      0659 1 |  
: 89      0660 1 | EXTERNAL REFERENCES:  
: 90      0661 1 |  
: 91      0662 1 |     In the routine
```

```

: 93      0663 1 %SBTTL 'EDT$$FMT_CHWID - compute formatted character width'
: 94      0664 1
: 95      0665 1 GLOBAL ROUTINE EDT$$FMT_CHWID (           ! Compute formatted character width
: 96      0666 1     CHAR,                               ! Character whose width to compute
: 97      0667 1     COLUMN                             ! Current column (for tabs)
: 98      0668 1     ) =
: 99      0669 1
: 100     0670 1 !++
: 101     0671 1 !FUNCTIONAL DESCRIPTION:
: 102     0672 1
: 103     0673 1     This routine computes the number of character positions a character would
: 104     0674 1     take to display at a specified column.
: 105     0675 1
: 106     0676 1 ! FORMAL PARAMETERS:
: 107     0677 1
: 108     0678 1     CHAR                               The character to be displayed.
: 109     0679 1
: 110     0680 1     COLUMN                             The starting column. (0 is the first column on a line)
: 111     0681 1
: 112     0682 1 ! IMPLICIT INPUTS:
: 113     0683 1
: 114     0684 1     EDT$$A_US_TXT
: 115     0685 1     EDT$$G_EIGHT_BIT
: 116     0686 1
: 117     0687 1 ! IMPLICIT OUTPUTS:
: 118     0688 1
: 119     0689 1     NONE
: 120     0690 1
: 121     0691 1 ! ROUTINE VALUE:
: 122     0692 1
: 123     0693 1     The number of character positions the character would occupy.
: 124     0694 1
: 125     0695 1 ! SIDE EFFECTS:
: 126     0696 1
: 127     0697 1     NONE
: 128     0698 1
: 129     0699 1 !--
: 130     0700 1
: 131     0701 2     BEGIN
: 132     0702 2
: 133     0703 2     EXTERNAL
: 134     0704 2
: 135     L 0705 2 %IF SUPPORT_VT220
: 136     0706 2 %THEN
: 137     0707 2     EDT$$G_EIGHT_BIT,                ! 1 = this is an eight-bit terminal
: 138     0708 2     EDT$$B_CHAR_INFO : BLOCKVECTOR [256, 1, BYTE], ! Table of information about characters
: 139     U 0709 2 %ELSE
: 140     U 0710 2     EDT$$B_CHAR_INFO : BLOCKVECTOR [128, 1, BYTE], ! Table of information about characters
: 141     0711 2 %FI
: 142     0712 2
: 143     0713 2     EDT$$A_CHAR_NAMES,                ! Names of some characters
: 144     0714 2     EDT$$K_CHAR_NAMES_LEN,            ! Length of the names table
: 145     0715 2     EDT$$A_US_TXT : VECTOR;
: 146     0716 2
: 147     0717 2     LOCAL
: 148     0718 2     C;
: 149     0719 2
```

```

: 150      0720      2      C = .CHAR;
: 151      0721      2
: 152      0722      3      %IF ( NOT SUPPORT_VT220)
: 153      0723      2      %THEN
: 154      0724      2      C = .C AND %X'7F';
: 155      0725      2      %FI
: 156      0726      2
: 157      0727      2      SELECTONE .C OF
: 158      0728      2      SET
: 159      0729      2
: 160      0730      2      [ASC_K_TAB] :
: 161      0731      2      RETURN (8 - (.COLUMN MOD 8));
: 162      0732      2
: 163      0733      2      [ASC_K_FF] :
: 164      0734      2      BEGIN
: 165      0735      2
: 166      0736      2      LOCAL
: 167      0737      2      LEN;
: 168      0738      2
: 169      0739      2      LEN = CH$RCHAR (.EDT$$A_US_TXT [1]);
: 170      0740      2      RETURN (.LEN);
: 171      0741      2      END;
: 172      0742      2
: 173      0743      2      [OTHERWISE] :
: 174      0744      2      BEGIN
: 175      0745      2
: 176      0746      2      LOCAL
: 177      0747      2      LEN;
: 178      0748      2
: 179      0749      2      CASE .EDT$$B_CHAR_INFO [.C, 0, 2, 2, 0] FROM 0 TO 3 OF
: 180      0750      2      SET
: 181      0751      2
: 182      0752      2      [0] :
: 183      0753      2      RETURN (1);          ! Normal character
: 184      0754      2
: 185      0755      2      [1] :
: 186      0756      2      RETURN (2);          ! ^c
: 187      0757      2
: 188      0758      2      [2] :
: 189      0759      2      BEGIN          ! <name>
: 190      0760      2
: 191      0761      2      LOCAL
: 192      0762      2      REP_PTR,
: 193      0763      2      REP_CHAR;
: 194      0764      2
: 195      L 0765      4      %IF SUPPORT_VT220
: 196      0766      4      %THEN
: 197      0767      4
: 198      0768      5      IF (.EDT$$G_EIGHT_BIT AND (.C GEQ %X'A0'))
: 199      0769      4      THEN
: 200      0770      5      RETURN (1)
: 201      0771      4      ELSE
: 202      0772      4      %FI
: 203      0773      4
: 204      0774      5      BEGIN
: 205      0775      5      REP_PTR = CH$PLUS (CH$FIND_CH (EDT$$K_CHAR_NAMES_LEN, EDT$$A_CHAR_NAMES, .C), 1);
: 206      0776      5      LEN = 2;

```

```

: 207      0777 5      REP_CHAR = CH$RCHAR_A (REP_PTR);
: 208      0778 5
: 209      0779 5      WHILE ((.REP_CHAR GEQ %X'20') AND (.REP_CHAR LEQ %X'7E')) DO
: 210      0780 6      BEGIN
: 211      0781 6      LEN = .LEN + 1;
: 212      0782 6      REP_CHAR = CH$RCHAR_A (REP_PTR);
: 213      0783 5      END;
: 214      0784 5
: 215      0785 5      RETURN (.LEN);
: 216      0786 4      END;
: 217      0787 4
: 218      0788 4      END;
: 219      0789 4
: 220      0790 4      [3] : RETURN (5);
: 221      0791 4      TES;
: 222      0792 4
: 223      0793 4
: 224      0794 4      END;
: 225      0795 4      TES;
: 226      0796 4
: 227      0797 1      END;

```

! of routine EDT\$\$FMT_CHWID

```

.TITLE EDT$FCOLINC EDT$FCOLINC - compute formatted cha
       racter width
.IDENT  \V04-000\
.EXTRN EDT$$G_EIGHT_BIT
.EXTRN EDT$$B_CHAR_INFO
.EXTRN EDT$$A_CHAR_NAMES
.EXTRN EDT$$K_CHAR_NAMES_LEN
.EXTRN EDT$$A_US_TXT

```

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

```

.ENTRY EDT$$FMT_CHWID, Save R2,R3
MOVL  CHAR, C
CPL  C, #9
BNEQ 1$
EMUL #1, COLUMN, #0, -(SP)
EDIV #8, (SP)+, R0, R0
SUBL3 R0, #8, R0
RET
CPL  C, #12
BNEQ 2$
MOVL EDT$$A_US_TXT+4, R0
MOVZBL (R0), [EN]
RET
EXTZV #2, #2, EDT$$B_CHAR_INFO[C], R3
CASEL R3, #0, #3
.WORD 6$-3$, -
      4$-3$, -
      5$-3$, -
      10$-3$
BRB 6$
MOVL #2, R0
RET

```

```

000C 00000
51 04 AC D0 00002
09 51 D1 00006
7E 00 08 AC 01 7A 0000B
50 50 8E 08 7B 00011
50 50 08 50 C3 00016
0C 51 D1 0001B 1$:
08 12 0001E
50 00000000G 00 D0 00020
50 60 9A 00027
53 00000000G0041 02 02 EF 0002B 2$:
00 03 00 53 CF 00035
004F 000E 000A 001E 00039 3$:
50 14 11 00041
02 D0 00043 4$:
04 00046

```

```

: 0665
: 0720
: 0730
: 0731
: 0733
: 0739
: 0740
: 0791
:

```


EDT\$FCOLINC
V04-000

EDT\$FCOLINC - compute formatted character width F 10
EDT\$\$FMT_CHWID - compute formatted character w 14-Sep-1984 00:19:37
14-Sep-1984 12:23:04

VAX-11 Bliss-32 V4.0-742 Page 7
DISK\$VMSMASTER:[EDT.SRC]FCOLINC.BLI;1 (4)

: 231 0800 1 END
: 232 0801 1
: 233 0802 0 ELUDOM

! of module EDT\$FCOLINC

PSECT SUMMARY

Name Bytes Attributes
_EDT\$CODE 140 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 | 2 | 0 | 40 | 00:00.2 |
| -\$255\$DUA28:[EDT.SRC]PSECTS.L32;1 | 2 | 1 | 50 | 7 | 00:00.1 |
| -\$255\$DUA28:[EDT.SRC]SUPPORTS.L32;1 | 2 | 1 | 50 | 5 | 00:00.1 |
| -\$255\$DUA28:[EDT.SRC]TRANSLATE.L32;1 | 6 | 0 | 0 | 57 | 00:00.1 |

COMMAND QUALIFIERS

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

: Size: 140 code + 0 data bytes
: Run Time: 00:13.0
: Elapsed Time: 00:16.3
: Lines/CPU Min: 3712
: Lexemes/CPU-Min: 10694
: Memory Used: 78 pages
: Compilation Complete

