C 10

```
EEEEEEEEEEEEEEE  DDDDDDDDDDDD     TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD     TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD     TTTTTTTTTTTTTTT
EEE              DDD         DDD        TTT
EEE              DDD         DDD        TTT
EEE              DDD         DDD        TTT
EEE              DDD         DDD        TTT
EEE              DDD         DDD        TTT
EEE              DDD         DDD        TTT
EEEEEEEEEEEE     DDD         DDD        TTT
EEEEEEEEEEEE     DDD         DDD        TTT
EEEEEEEEEEEE     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
EEEEEEEEEEEEEEE  DDDDDDDDDDDD           TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD           TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD           TTT
```

```
EX

Mod
---
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
SY
LB
LI
```

```
**FILE**ID**WFREAFWD                                   D 12



WW      WW  FFFFFFFFFF  RRRRRRRR    EEEEEEEEEE    AAAAAA    FFFFFFFFFF  WW      WW  DDDDDDDD
WW      WW  FFFFFFFFFF  RRRRRRRR    EEEEEEEEEE    AAAAAA    FFFFFFFFFF  WW      WW  DDDDDDDD
WW      WW  FF          RR      RR  EE          AA      AA  FF          WW      WW  DD      DD
WW      WW  FF          RR      RR  EE          AA      AA  FF          WW      WW  DD      DD
WW      WW  FF          RR      RR  EE          AA      AA  FF          WW      WW  DD      DD
WW      WW  FF          RR      RR  EE          AA      AA  FF          WW      WW  DD      DD
WW      WW  FFFFFFFF    RRRRRRRR    EEEEEEEE    AA      AA  FFFFFFFF    WW      WW  DD      DD
WW      WW  FFFFFFFF    RRRRRRRR    EEEEEEEE    AA      AA  FFFFFFFF    WW      WW  DD      DD
WW  WW  WW  FF          RR  RR      EE          AAAAAAAAAA  FF          WW  WW  WW  DD      DD
WW  WW  WW  FF          RR  RR      EE          AAAAAAAAAA  FF          WW  WW  WW  DD      DD
WWWW  WWWW  FF          RR    RR    EE          AA      AA  FF          WWWW  WWWW  DD      DD      ....
WWWW  WWWW  FF          RR    RR    EE          AA      AA  FF          WWWW  WWWW  DD      DD      ....
WW      WW  FF          RR      RR  EEEEEEEEEE  AA      AA  FF          WW      WW  DDDDDDDD        ....
WW      WW  FF          RR      RR  EEEEEEEEEE  AA      AA  FF          WW      WW  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
```

EDTS
V04-

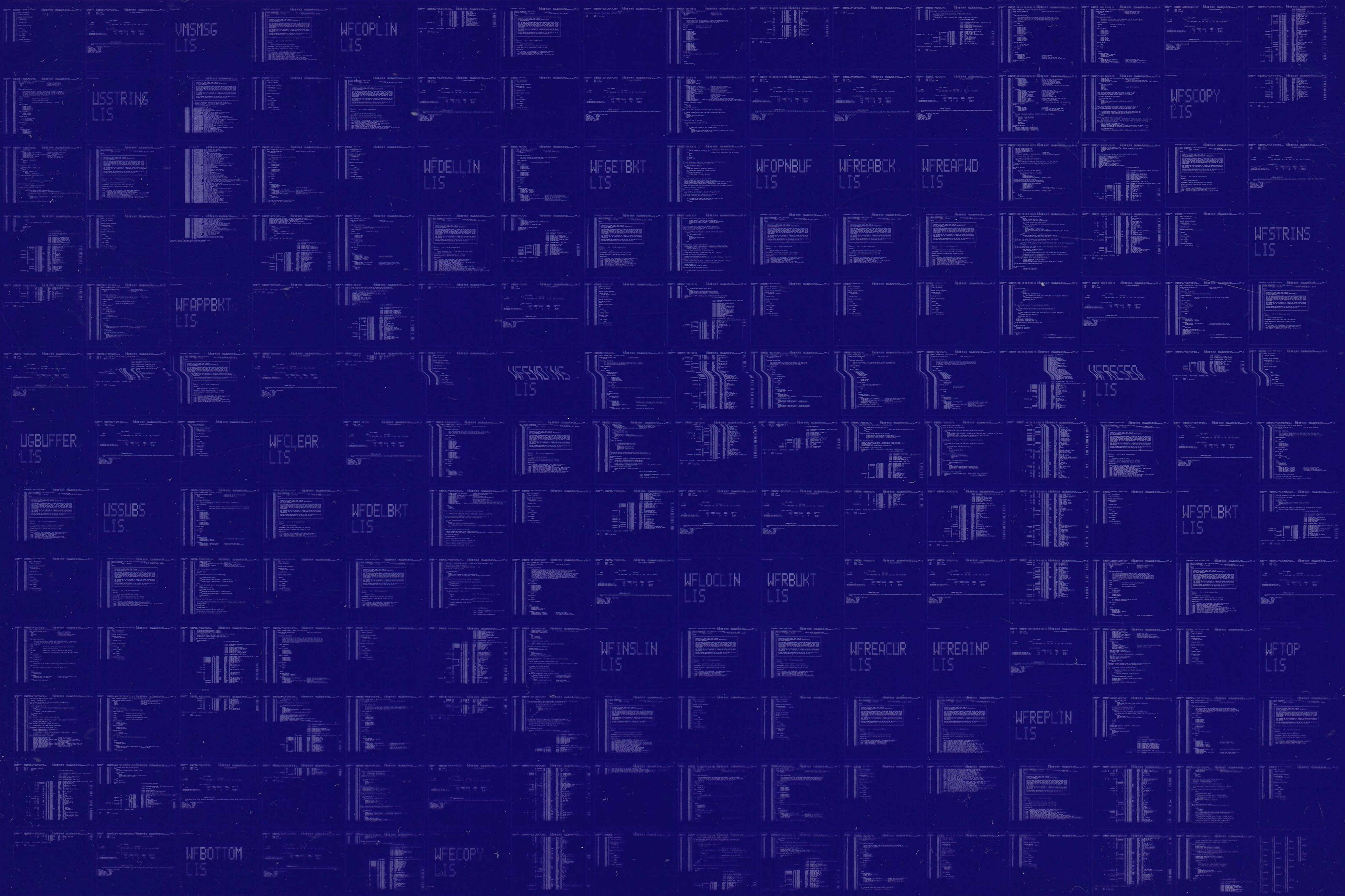
```
0 %TITLE 'EDT$WFREAFWD - read the next line'
0 MODULE EDT$WFREAFWD (                                      ! Read the next line
0               IDENT = 'V04-000'                                     ! File: WFREAFWD.BLI Edit: STS1008
0               ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1 
1 !++
1 ! FACILITY:      EDT -- The DEC Standard Editor
1 !
1 ! ABSTRACT:
1 !
1 !       Read the next line in the forward direction.
1 !
1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
1 !
1 ! AUTHOR: Bob Kushlis, CREATION DATE: October 16, 1978
1 !
1 ! MODIFIED BY:
1 !
1 ! 1-001 - Original.  DJS 23-Feb-1981.  This module was created by
1 !       extracting routine EDT$$RD_NXTLN  from module EDTWF.
1 ! 1-002 - Regularized the headers.  JBS 25-Feb-1981
1 ! 1-003 - Fix module name.  JBS 19-Mar-1981
1 ! 1-004 - Change EOB_LINE to EDT$$Z_EOB_LN .  JBS 31-Mar-1981
1 ! 1-005 - Correct a typo in a subtitle.  JBS 02-Jun-1981
1 ! 1-006 - Change index for line numbers from 10 to 15.  SMB 18-Jan-1982
1 ! 1-007 - Remove EDT$$SET_WKLN.  JBS 14-Sep-1982
1 ! 1-008 - Modify to use new 48 bit macro. STS 01-Oct-1982
1 !--
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$$RD_NXTLN;
;     65            0502  1
;     66            0503  1 !
;     67            0504  1 ! INCLUDE FILES:
;     68            0505  1 !
;     69            0506  1
;     70            0507  1 REQUIRE 'EDTSRC:EDTREQ';
;     71            0642  1
;     72            0643  1 !
;     73            0644  1 ! MACROS:
;     74            0645  1 !
;     75            0646  1 !       NONE
;     76            0647  1 !
;     77            0648  1 ! EQUATED SYMBOLS:
;     78            0649  1 !
;     79            0650  1 !       NONE
;     80            0651  1 !
;     81            0652  1 ! OWN STORAGE:
;     82            0653  1 !
;     83            0654  1 !       NONE
;     84            0655  1 !
;     85            0656  1 ! EXTERNAL REFERENCES:
;     86            0657  1 !
;     87            0658  1 !       In the routine
;     89          0659  1 %SBTTL 'EDT$$RD_NXTLN  - read the next lire'
;     90          0660  1
;     91          0661  1 GLOBAL ROUTINE EDT$$RD_NXTLN                          ! Read the next line
;     92          0662  1     =
;     93          0663  1
;     94          0664  1 !++
;     95          0665  1 ! FUNCTIONAL DESCRIPTION:
;     96          0666  1 !
;     97          0667  1 !       Read the next line in the forward direction.  The line following the
;     98          0668  1 !       current line becomes the new line.  This routine may have the effect
;     99          0669  1 !       of reading a line from the input file.  If we are already at the end
;    100          0670  1 !       of the buffer, then return a 0 otherwise return a 1.
;    101          0671  1 !
;    102          0672  1 ! FORMAL PARAMETERS:
;    103          0673  1 !
;    104          0674  1 !       NONE
;    105          0675  1 !
;    106          0676  1 ! IMPLICIT INPUTS:
;    107          0677  1 !
;    108          0678  1 !       EDT$$G_EXITD
;    109          0679  1 !       EDT$$A_WK_BUK
;    110          0680  1 !       EDT$$G_WK_CURBUK
;    111          0681  1 !       EDT$$A_WK_LN
;    112          0682  1 !       EDT$$Z_EOB_LN
;    113          0683  1 !       EDT$$L_LNO0
;    114          0684  1 !
;    115          0685  1 ! IMPLICIT OUTPUTS:
;    116          0686  1 !
;    117          0687  1 !       EDT$$A_CUR_BUF
;    118          0688  1 !       EDT$$A_WK_LN
;    119          0689  1 !
;    120          0690  1 ! ROUTINE VALUE:
;    121          0691  1 !
;    122          0692  1 !       1               Not at end of buffer
;    123          0693  1 !       0               At end of buffer
;    124          0694  1 !
;    125          0695  1 ! SIDE EFFECTS:
;    126          0696  1 !
;    127          0697  1 !       NONE
;    128          0698  1 !
;    129          0699  1 !--
;    130          0700  1
;    131          0701  2     BEGIN
;    132          0702  2
;    133          0703  2     EXTERNAL ROUTINE
;    134          0704  2         EDT$$WF_MAKECUR : NOVALUE,
;    135          0705  2         EDT$$RD_ILN;
;    136          0706  2
;    137          0707  2     EXTERNAL
;    138          0708  2         EDT$$A_CUR_BUF : REF TBCB_BLOCK,        ! Current text buffer control block
;    139          0709  2         EDT$$G_EXITD,                           ! Exit flag (on if we are exiting)
;    140          0710  2         EDT$$A_WK_BUK :                         ! Pointer to current bucket
;    141          0711  2             REF BLOCK [WF_BUKT_SIZE, BYTE] FIELD (WFB_FIELDS),
;    142          0712  2         EDT$$G_WK_CURBUK,                       ! Number of the current bucket
;    143          0713  2         EDT$$A_WK_LN : REF LIN_BLOCK,           ! Pointer to current line
;    144          0714  2         EDT$$Z_EOB_LN,
;    145          0715  2         EDT$$L_LNO0 : LNOVECTOR [14];
;   146     0716  2
;   147     0717  2        EDT$$A_CUR_BUF [TBCB_CHAR_POS] = 0;
;   148     0718  2    !+
;   149     0719  2    ! Point to the next line in the bucket.
;   150     0720  2    !-
;   151     0721  2
;   152     0722  3        IF (.EDT$$A_CUR_BUF [TBCB_LINE_ADDR] NEQA .EDT$$A_WK_BUK [WFB_END])
;   153     0723  2        THEN
;   154     0724  3            BEGIN
;   155     0725  3            EDT$$A_CUR_BUF [TBCB_LINE_ADDR] = .EDT$$A_CUR_BUF [TBCB_LINE_ADDR] + .EDT$$A_WK_LN [LIN_LENGTH] +
;   156     0726  3            LIN_FIXED_SIZE + 1;
;   157     0727  3            EDT$$A_WK_LN = CH$PTR (.EDT$$A_WK_BUK, .EDT$$A_CUR_BUF [TBCB_LINE_ADDR]);
;   158     0728  3            ADDLINE (NUMBER_ONE, EDT$$A_CUR_BUF [TBCB_CUR_LIN]);
;   159     0729  2            END;
;   160     0730  2
;   161     0731  2    !+
;   162     0732  2    ! If this is out of the scope of the bucket, then we better read the
;   163     0733  2    ! next bucket.
;   164     0734  2    !-
;   165     0735  2
;   166     0736  3        IF (.EDT$$A_CUR_BUF [TBCB_LINE_ADDR] GEQA .EDT$$A_WK_BUK [WFB_END])
;   167     0737  2        THEN
;   168     0738  2
;   169     0739  3            IF (.EDT$$A_WK_BUK [WFB_NEXT_BUKT] EQL 0)
;   170     0740  2            THEN
;   171     0741  3                BEGIN
;   172     0742  3
;   173     0743  4                IF (.EDT$$G_EXITD NEQ 0)
;   174     0744  3                THEN
;   175     0745  4                    BEGIN
;   176     0746  4    !+
;   177     0747  4    ! Return zero so we don't read any more.
;   178     0748  4    !-
;   179     0749  4                    EDT$$A_WK_LN = EDT$$Z_EOB_LN;
;   180     0750  4                    RETURN (0);
;   181     0751  3                    END;
;   182     0752  3
;   183     0753  3    !+
;   184     0754  3    ! End of buffer, try reading the next record.
;   185     0755  3    !-
;   186     0756  3                RETURN (EDT$$RD_ILN ());
;   187     0757  3                END
;   188     0758  2            ELSE
;   189     0759  3                BEGIN
;   190     0760  3                EDT$$WF_MAKECUR (.EDT$$A_WK_BUK [WFB_NEXT_BUKT]);
;   191     0761  3                EDT$$A_CUR_BUF [TBCB_LINE_ADDR] = WFB_FIXED_SIZE;
;   192     0762  3                EDT$$A_CUR_BUF [TBCB_CUR_BUKT] = .EDT$$G_WK_CURBUK;
;   193     0763  2                END;
;   194     0764  2
;   195     0765  2    !+
;   196     0766  2    ! Update the current line pointer.
;   197     0767  2    !-
;   198     0768  2        EDT$$A_WK_LN = CH$PTR (.EDT$$A_WK_BUK, .EDT$$A_CUR_BUF [TBCB_LINE_ADDR]);
;   199     0769  3        RETURN (1)
;   200     0770  1        END;                                        ! of routine EDT$$RD_NXTLN
```

```
                                             .TITLE  EDT$WFREAFWD EDT$WFREAFWD - read the next line
                                             .IDENT  \V04-000\

                                             .EXTRN  EDT$$WF_MAKECUR
                                             .EXTRN  EDT$$RD_ILN, EDT$$A_CUR_BUF
                                             .EXTRN  EDT$$G_EXITD, EDT$$A_WK_BUK
                                             .EXTRN  EDT$$G_WK_CURBUK
                                             .EXTRN  EDT$$A_WK_LN, EDT$$Z_EOB_LN
                                             .EXTRN  EDT$$L_LN00

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

                             003C 00000      .ENTRY  EDT$$RD_NXTLN, Save R2,R3,R4,R5            ; 0661
             55 00000000G  00  9E 00002      MOVAB   EDT$$A_WK_BUK, R5                          ;
             54 00000000G  00  9E 00009      MOVAB   EDT$$A_CUR_BUF, R4                         ;
             53 00000000G  00  9E 00010      MOVAB   EDT$$A_WK_LN, R3                           ;
             50            64  D0 00017      MOVL    EDT$$A_CUR_BUF, R0                         ; 0717
                       0C  A0  B4 0001A      CLRW    12(R0)                                     ;
             51            65  D0 0001D      MOVL    EDT$$A_WK_BUK, R1                          ; 0722
         04  A1            60  D1 00020      CMPL    (R0), 4(R1)                                ;
                           19  13 00024      BEQL    1$                                         ;
             52            63  D0 00026      MOVL    EDT$$A_WK_LN, R2                           ; 0725
             52            62  9A 00029      MOVZBL  (R2), R2                                   ;
             52            60  C0 0002C      ADDL2   (R0), R2                                   ;
             60        08  A2  9E 0002F      MOVAB   8(R2), (R0)                                ; 0726
63           51            60  C1 00033      ADDL3   (R0), R1, EDT$$A_WK_LN                     ; 0727
                       06  A0  D6 00037      INCL    6(R0)                                      ; 0728
                           03  12 0003A      BNEQ    1$                                         ;
                       0A  A0  B6 0003C      INCW    10(R0)                                     ;
         04  A1            60  D1 0003F 1$:  CMPL    (R0), 4(R1)                                ; 0736
                           37  1F 00043      BLSSU   4$                                         ;
                       02  A1  B5 00045      TSTW    2(R1)                                      ; 0739
                           19  12 00048      BNEQ    3$                                         ;
                0000000G   00  D5 0004A      TSTL    EDT$$G_EXITD                               ; 0743
                           09  13 00050      BEQL    2$                                         ;
             63 00000000G  00  9E 00052      MOVAB   EDT$$Z_EOB_LN, EDT$$A_WK_LN                ; 0749
                           2C  11 00059      BRB     5$                                         ; 0750
   00000000G 00            00  FB 0005B 2$:  CALLS   #0, EDT$$RD_ILN                            ; 0756
                               04 00062      RET                                                ;
             7E        02  A1  3C 00063 3$:  MOVZWL  2(R1), -(SP)                               ; 0760
   00000000G 00            01  FB 00067      CALLS   #1, EDT$$WF_MAKECUR                        ;
             50            64  D0 0006E      MOVL    EDT$$A_CUR_BUF, R0                         ; 0761
             60            08  D0 00071      MOVL    #8, (R0)                                   ;
         04  A0 00000000G  00  B0 00074      MOVW    EDT$$G_WK_CURBUK, 4(R0)                    ; 0762
             50            64  D0 0007C 4$:  MOVL    EDT$$A_CUR_BUF, R0                         ; 0768
63           65            60  C1 0007F      ADDL3   (R0), EDT$$A_WK_BUK, EDT$$A_WK_LN          ;
             50            01  D0 00083      MOVL    #1, R0                                     ; 0769
                               04 00086      RET                                                ;
                           50  D4 00087 5$:  CLRL    R0                                         ; 0770
                               04 00089      RET                                                ;

; Routine Size:  138 bytes,    Routine Base:  _EDT$CODE + 0000


;  201          0771  1
;  202          0772  1 !<BLF/PAGE>
;  204           0773  1 END                                                 ! of module EDT$WFREAFWD
;  205           0774  1
;  206           0775  0 ELUDOM




;                                         PSECT SUMMARY
;
;        Name                          Bytes                          Attributes
;
;   _EDT$CODE                               138  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)



;                                  Library Statistics
;
;                                              -------- Symbols --------          Pages          Processing
;        File                                  Total      Loaded   Percent        Mapped         Time
;
;   _$255$DUA28:[EDT.SRC]EDT.L32;1                 377         42        11            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$:WFREAFWD/OBJ=OBJ$:WFREAFWD MSRC$:WFREAFWD.BLI/UPDATE=(ENH$:WFRE
;        AFWD)

; Size:            138 code + 0 data bytes
; Run Time:           00:13.3
; Elapsed Time:       00:15.8
; Lines/CPU Min:        3496
; Lexemes/CPU-Min: 13069
; Memory Used:  93 pages
; Compilation Complete
```

0141 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



VMSMSG LIS

WFCOPLIN LIS

USSTRING LIS

WFSCOPY LIS

WFDELLIN LIS

WFGETBKT LIS

WFOPNBUF LIS

WFREABCK LIS

WFREAFWD LIS

WFSTRINS LIS

WFAPPBKT LIS

WFENDINS LIS

[illegible] LIS

UGBUFFER LIS

WFCLEAR LIS

USSUBS LIS

WFDELBKT LIS

WFSPLBKT LIS

WFLOCLIN LIS

WFRBUKT LIS

WFINSLIN LIS

WFREACUR LIS

WFREAINP LIS

WFTOP LIS

WFREPLIN LIS

WFBOTTOM LIS

WFECOPY LIS