```
EEEEEEEEEEEEEEE  DDDDDDDDDDD     FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFFFFFFFFFFFFFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFFFFFFFFFFFFFF
EEE              DDD         DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEEEEEEEEEEE     DDD          DDD FFFFFFFFFFFF
EEEEEEEEEEEE     DDD          DDD FFFFFFFFFFFF
EEEEEEEEEEEE     DDD          DDD FFFFFFFFFFFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEE              DDD          DDD FFF
EEEEEEEEEEEEEEE  DDDDDDDDDDD     FFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFF
EEEEEEEEEEEEEEE  DDDDDDDDDDDD    FFF
```

```
_$
Va
--
00
00
00
00
00
00
00
00
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
7F
```

FILEID**EDFSHOW

```
EEEEEEEEEE  DDDDDDDD   FFFFFFFFFF   SSSSSSSS  HH      HH   OOOOOO   WW      WW
EEEEEEEEEE  DDDDDDDD   FFFFFFFFFF   SSSSSSSS  HH      HH   OOOOOO   WW      WW
EE          DD      DD FF           SS        HH      HH  OO      OO WW      WW
EE          DD      DD FF           SS        HH      HH  OO      OO WW      WW
EE          DD      DD FF           SS        HH      HH  OO      OO WW      WW
EE          DD      DD FF           SS        HH      HH  OO      OO WW      WW
EEEEEEEE    DD      DD FFFFFFF       SSSSSS   HHHHHHHHHH  OO      OO WW      WW
EEEEEEEE    DD      DD FFFFFFF       SSSSSS   HHHHHHHHHH  OO      OO WW      WW
EE          DD      DD FF                 SS  HH      HH  OO      OO WW  WW  WW
EE          DD      DD FF                 SS  HH      HH  OO      OO WW  WW  WW
EE          DD      DD FF                 SS  HH      HH  OO      OO WWWW  WWWW     ....
EE          DD      DD FF                 SS  HH      HH  OO      OO WWWW  WWWW     ....
EEEEEEEEEE  DDDDDDDD   FF           SSSSSSSS  HH      HH   OOOOOO   WW      WW     ....
EEEEEEEEEE  DDDDDDDD   FF           SSSSSSSS  HH      HH   OOOOOO   WW      WW     ....

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            [ IDENT ('V04-000'),
0002        { ++
0003        ******************************************************************************
0004        **                                                                          *
0005        **  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0006        **  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0007        **  ALL RIGHTS RESERVED.                                                    *
0008        **                                                                          *
0009        **  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0010        **  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0011        **  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0012        **  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0013        **  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0014        **  TRANSFERRED.                                                            *
0015        **                                                                          *
0016        **  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0017        **  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0018        **  CORPORATION.                                                            *
0019        **                                                                          *
0020        **  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0021        **  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0022        **                                                                          *
0023        **                                                                          *
0024        ******************************************************************************
0025
0026
0027
0028
0029        FACILITY:       VAX/VMS EDF (EDIT/FDL) UTILITY
0030
0031        ABSTRACT:       This facility is used to create, modify, and optimize
0032                        FDL specification files.
0033
0034        ENVIRONMENT:    NATIVE/USER MODE
0035
0036        AUTHOR:         Ken F. Henderson Jr.
0037
0038        CREATION DATE:  27-Mar-1981
0039
0040        MODIFIED BY:
0041
0042                        V03-08  KFH0008         Ken Henderson           10 Sep 1983
0043                                Support for named UICs
0044
0045                        V03-007 KFH0007         Ken Henderson            8 Aug 1983
0046                                Changes for seperate compilation.
0047
0048                        V03-006 KFH0006         Ken Henderson           30 Jul 1983
0049                                Added never_RU_journal.
0050
0051                        V03-005 KFH0005         Ken Henderson           14 Apr 1983
0052                                Fixed SHOW_PRIMARY_SECTION routine.
0053                                Fixed CHECK_QUOTES routine.
0054
0055                        V03-004 KFH0004         Ken Henderson           22 Nov 1982
0056                                Removed the EDF$_INTSWERR numbers.
0057
```

```
0058                    V03-003 KFH0003         Ken Henderson           8 Sept 1982
0059                            Added CHECK_QUOTES routine.
0060
0061                    V03-002 KFH0002         Ken Henderson           23-Mar-1982
0062                            Reorderred the EDF$_INTSWERR numbers.
0063
0064                    V03-001 KFH0001         Ken Henderson           17-Mar-1982
0065                            Modified SHOW_CURRENT to fix FT2 QARs
0066                            440,449
0067
0068            -- }
```

```
0070            ENVIRONMENT ('LIB$:EDFSHOW'),
0071
0072            INHERIT (
0073
0074            'SYS$LIBRARY:STARLET',
0075            'SHRLIB$:FDLPARDEF',
0076            'LIB$:EDFSDLMSG',
0077            'LIB$:EDFSTRUCT',
0078            'LIB$:EDFCONST',
0079            'LIB$:EDFTYPE',
0080            'LIB$:EDFVAR',
0081            'LIB$:EDFEXTERN',
0082            'LIB$:EDFCHF',
0083            'LIB$:EDFUTIL',
0084            'LIB$:EDFASK'
0085
0086            )]
0087
0088            MODULE EDFSHOW (INPUT,OUTPUT);
```

```
0090            { ++
0091
0092            CHECK_QUOTES -- Bracket the output string with quotes.
0093
0094            This routine scans for quotes that bracket a string-valued output secondary.
0095            If they aren't there, it puts on "s. If there is a " in the string, it puts
0096            on 's. If there are both in the string, then it puts on "s (doubling embedded
0097            "s).
0098
0099            CALLING SEQUENCE:
0100
0101            CHECK_QUOTES (DESC);
0102
0103            INPUT PARAMETERS:
0104
0105            DESC - the character string to scan.
0106
0107            IMPLICIT INPUTS:
0108
0109            APOSTROPHE
0110
0111            OUTPUT PARAMETERS:
0112
0113            DESC - the scanned character string
0114
0115            IMPLICIT OUTPUTS:
0116
0117            none
0118
0119            ROUTINES CALLED:
0120
0121            none
0122
0123            ROUTINE VALUE:
0124
0125            none
0126
0127            SIGNALS:
0128
0129            none
0130
0131            SIDE EFFECTS:
0132
0133            none
0134
0135            -- }
```

```
0137        PROCEDURE CHECK_QUOTES (VAR DESC: DESCRIPTOR);
0138
0139        VAR
0140            SCAN_INDEX  : INTEGER;
0141            INDEX       : INTEGER;
0142            QUOTE_FOUND : BOOLEAN;
0143            APOST_FOUND : BOOLEAN;
0144            QCHAR       : CHAR;
0145            QUOTES      : CHAR;
0146
0147        BEGIN
0148
0149            QUOTES      := CHR (34);
0150
0151            { +
0152            Non-null strings are the most interesting.
0153            - }
0154            IF DESC.DSC$W_LENGTH <> 0 THEN
0155
0156            BEGIN
0157
0158                { +
0159                Don't disturb a string that already is enclosed in (')s or (")s.
0160                - }
0161                IF NOT (
0162                    (
0163                    (DESC.DSC$A_POINTER^[1] = APOSTROPHE)
0164                    AND
0165                    (DESC.DSC$A_POINTER^[DESC.DSC$W_LENGTH] = APOSTROPHE)
0166                    )
0167                OR
0168                    (
0169                    (DESC.DSC$A_POINTER^[1] = QUOTES)
0170                    AND
0171                    (DESC.DSC$A_POINTER^[DESC.DSC$W_LENGTH] = QUOTES)
0172                    )
0173                ) THEN
0174
0175                BEGIN
0176
0177                    { +
0178                    First, see if there are any nasty embedded quotes or apostrophes.
0179                    - }
0180                    QUOTE_FOUND := FALSE;
0181                    APOST_FOUND := FALSE;
0182
0183                    FOR INDEX   := 1 TO DESC.DSC$W_LENGTH DO
0184
0185                    BEGIN
0186
0187                        IF DESC.DSC$A_POINTER^[INDEX] = QUOTES THEN
0188
0189                            QUOTE_FOUND := TRUE;
0190
0191                        IF DESC.DSC$A_POINTER^[INDEX] = APOSTROPHE THEN
0192
0193                            APOST_FOUND := TRUE;
```

```
0194
0195                    END;
0196
0197                    { +
0198                    Just bracket the string with quotes if no quotes are found,
0199                    or if just bracket it with apostrophes if no apostrophes are
0200                    found.
0201                    - }
0202                    IF QUOTE_FOUND THEN
0203
0204                    BEGIN
0205
0206                        IF APOST_FOUND THEN
0207
0208                        BEGIN
0209
0210                            SCAN_INDEX  := DESC.DSC$W_LENGTH;
0211
0212                            REPEAT
0213
0214                                IF DESC.DSC$A_POINTER^[SCAN_INDEX] = QUOTES THEN
0215
0216                                BEGIN
0217
0218                                    FOR INDEX := DESC.DSC$W_LENGTH DOWNTO SCAN_INDEX DO
0219
0220                                        DESC.DSC$A_POINTER^[INDEX+1]
0221                                                        := DESC.DSC$A_POINTER^[INDEX];
0222
0223                                    IF DESC.DSC$W_LENGTH < 254 THEN
0224
0225                                        DESC.DSC$W_LENGTH := DESC.DSC$W_LENGTH + 1;
0226
0227                                END;
0228
0229                                SCAN_INDEX      := SCAN_INDEX - 1;
0230
0231                            UNTIL SCAN_INDEX < 1;
0232
0233                            QCHAR       := QUOTES;
0234
0235                        END     { IF TRUE APOST_FOUND (AND QUOTE_FOUND) -YUK }
0236
0237                        ELSE
0238
0239                        BEGIN
0240
0241                            QCHAR       := APOSTROPHE;
0242
0243                        END;    { IF FALSE APOST_FOUND }
0244
0245                    END         { IF TRUE QUOTE_FOUND }
0246
0247                    ELSE
0248
0249                    BEGIN
0250
```

```
0251                    QCHAR           := QUOTES;
0252
0253                END;        { IF FALSE QUOTE_FOUND }
0254
0255                { +
0256                Shift the string down one char.
0257                - }
0258                FOR INDEX   := DESC.DSC$W_LENGTH DOWNTO 1 DO
0259
0260                    DESC.DSC$A_POINTER^[INDEX+1]    := DESC.DSC$A_POINTER^[INDEX];
0261
0262                { +
0263                Update the length and add the quotes.
0264                - }
0265                DESC.DSC$W_LENGTH                   := DESC.DSC$W_LENGTH + 2;
0266                DESC.DSC$A_POINTER^[1]                      := QCHAR;
0267                DESC.DSC$A_POINTER^[DESC.DSC$W_LENGTH]      := QCHAR;
0268
0269            END;    { IF NOT (ALREADY QUOTED) }
0270
0271        END         { IF TRUE DESC.DSC$W_LENGTH <> 0 }
0272
0273        ELSE
0274
0275        BEGIN
0276
0277            STR$TRIM (DESC,EMPTY_STRING);
0278
0279        END;        { IF FALSE DESC.DSC$W_LENGTH <> 0 }
0280
0281    END;    { CHECK_QUOTES }
```

```
0283            { ++
0284
0285            SHOW_PRIMARY -- Output the text string for the primary.
0286
0287            This routine outputs the primary keyword.
0288
0289            CALLING SEQUENCE:
0290
0291            SHOW_PRIMARY;
0292
0293            INPUT PARAMETERS:
0294
0295            none
0296
0297            IMPLICIT INPUTS:
0298
0299            DEF_CURRENT
0300            PRIMARY_WIDTH
0301
0302            OUTPUT PARAMETERS:
0303
0304            none
0305
0306            IMPLICIT OUTPUTS:
0307
0308            FDL_DEST
0309
0310            ROUTINES CALLED:
0311
0312            none
0313
0314            ROUTINE VALUE:
0315
0316            none
0317
0318            SIGNALS:
0319
0320            none
0321
0322            SIDE EFFECTS:
0323
0324            The text is shown on the terminal or put in the file. Wherever FDL_DEST points.
0325
0326            -- }
```

```
0328            PROCEDURE SHOW_PRIMARY;
0329
0330            VAR
0331                TEMP_PRI    : PRIMARY_TYPE;
0332
0333            BEGIN
0334
0335                WITH DEF_CURRENT^ DO
0336
0337                BEGIN
0338
0339                    { +
0340                    Fetch the primary we're showing.
0341                    - }
0342                    TEMP_PRI        := PRIMARY;
0343
0344                    { +
0345                    Output it, using the correct width.
0346                    - }
0347                    WRITE (FDL_DEST,TEMP_PRI:PRIMARY_WIDTH[TEMP_PRI]);
0348
0349                    { +
0350                    If it's one with a PRINUM, put that out too.
0351                    - }
0352                    IF TEMP_PRI IN [ ANALYSIS_OF_AREA, ANALYSIS_OF_KEY, AREA, KEY ] THEN
0353
0354                        WRITE (FDL_DEST,' ',PRINUM:NUM_LEN(PRINUM));
0355
0356                    { +
0357                    If it's one with a string value, put that out too.
0358                    - }
0359                    IF TEMP_PRI IN [ IDENT, TITLE ] THEN
0360
0361                    BEGIN
0362
0363                        CHECK_QUOTES (STRING);
0364
0365                        IF (STRING.DSC$W_LENGTH > 0) THEN
0366
0367                            WRITE (FDL_DEST,'       ',STRING.DSC$A_POINTER^:
0368                            STRING.DSC$W_LENGTH);
0369
0370                    END;
0371
0372                END;        { WITH DEF_CURRENT^ }
0373
0374            END;    { SHOW_PRIMARY }
```

```
0376          { ++
0377
0378          SHOW_SECONDARY -- Output a text string for the secondary keyword.
0379
0380          This routine outputs a secondary keyword.
0381
0382          CALLING SEQUENCE:
0383
0384          SHOW_SECONDARY;
0385
0386          INPUT PARAMETERS:
0387
0388          none
0389
0390          IMPLICIT INPUTS:
0391
0392          DEF_CURRENT
0393          SECONDARY_WIDTH
0394
0395          OUTPUT PARAMETERS:
0396
0397          none
0398
0399          IMPLICIT OUTPUTS:
0400
0401          FDL_DEST
0402
0403          ROUTINES CALLED:
0404
0405          none
0406
0407          ROUTINE VALUE:
0408
0409          none
0410
0411          SIGNALS:
0412
0413          none
0414
0415          SIDE EFFECTS:
0416
0417          The secondary keyword is put into the file or onto the terminal. (FDL_DEST)
0418
0419          -- }
```

```
0421        PROCEDURE SHOW_SECONDARY;
0422
0423        VAR
0424            TEMP_SEC    : SECONDARY_TYPE;
0425
0426        BEGIN
0427
0428            WITH DEF_CURRENT^ DO
0429
0430            BEGIN
0431
0432                { +
0433                Fetch the secondary.
0434                - }
0435                TEMP_SEC        := SECONDARY;
0436
0437                { +
0438                Output the secondary if it's a simply structured one.
0439                - }
0440                IF NOT ( TEMP_SEC IN [ SEG_LENGTH, SEG_POSITION, SEG_TYPE ] ) THEN
0441
0442                BEGIN
0443
0444                    WRITE (FDL_DEST,'   ',TEMP_SEC:SECONDARY_WIDTH[TEMP_SEC]);
0445
0446                    { +
0447                    Put out extra tabs to compensate for short secondaries.
0448                    - }
0449                    IF (
0450                    (TEMP_SEC = POSITION$)
0451                    AND
0452                    (NOT (QUALIFIER IN [ FDL$C_ANYPOS, FDL$C_NOPOS ]))
0453                    ) THEN
0454
0455                        WRITE (FDL_DEST,'    ')
0456
0457                    ELSE IF SECONDARY_WIDTH[TEMP_SEC] < 8 THEN
0458
0459                        WRITE (FDL_DEST,'               ')
0460
0461                    ELSE IF SECONDARY_WIDTH[TEMP_SEC] < 16 THEN
0462
0463                        WRITE (FDL_DEST,'       ');
0464
0465            END
0466
0467            ELSE
0468
0469            { +
0470            Here for the complicated secondaries: SEGn_xxx
0471            - }
0472            BEGIN
0473
0474                IF TEMP_SEC = SEG_LENGTH THEN
0475
0476                    WRITE (FDL_DEST,'       SEG',SECNUM:1,'_LENGTH  ');
0477
```

```
0478                    { +
0479                    Now do the same for the SEGn_POSITION secondary.
0480                    - }
0481                    IF TEMP_SEC = SEG_POSITION THEN
0482
0483                        WRITE (FDL_DEST,'       SEG',SECNUM:1,'_POSITION        ');
0484
0485                    { +
0486                    Ditto for SEGn_TYPE.
0487                    - }
0488                    IF TEMP_SEC = SEG_TYPE THEN
0489
0490                        WRITE (FDL_DEST,'       TYPE            ');
0491
0492                { +
0493                SEGn_TYPE will NOT be supported until after version V3B
0494                - }
0495
0496                {               WRITE (FDL_DEST,'       SEG',SECNUM:1,'_TYPE    '); }
0497
0498                    END;
0499
0500                END;        { WITH DEF_CURRENT^ }
0501
0502            END;    { SHOW_SECONDARY }
```

```
0504            { ++
0505
0506            SHOW_QUALIFIER -- Output a text string for the qualifier keyword.
0507
0508            This routine outputs the qualifier keyword.
0509
0510            CALLING SEQUENCE:
0511
0512            SHOW_QUALIFIER;
0513
0514            INPUT PARAMETERS:
0515
0516            none
0517
0518            IMPLICIT INPUTS:
0519
0520            DEF_CURRENT
0521
0522            OUTPUT PARAMETERS:
0523
0524            none
0525
0526            IMPLICIT OUTPUTS:
0527
0528            FDL_DEST
0529
0530            ROUTINES CALLED:
0531
0532            none
0533
0534            ROUTINE VALUE:
0535
0536            none
0537
0538            SIGNALS:
0539
0540            none
0541
0542            SIDE EFFECTS:
0543
0544            The keyword is put into the file or onto the terminal. (FDL_DEST)
0545
0546            -- }
```

```
0548        PROCEDURE SHOW_QUALIFIER;
0549
0550        BEGIN
0551
0552            { +
0553            Output the qualifiers.
0554            - }
0555            CASE DEF_CURRENT^.QUALIFIER OF
0556
0557                FDL$C_ANYPOS :          WRITE (FDL_DEST,'       any_cylinder');
0558                FDL$C_CLUSPOS :         WRITE (FDL_DEST,'cluster');
0559                FDL$C_CYLPOS :          WRITE (FDL_DEST,'cylinder');
0560                FDL$C_FIDPOS :          WRITE (FDL_DEST,'file_ID');
0561                FDL$C_FNMPOS :          WRITE (FDL_DEST,'file_name');
0562                FDL$C_LOGPOS :          WRITE (FDL_DEST,'logical');
0563                FDL$C_VIRPOS :          WRITE (FDL_DEST,'virtual');
0564                FDL$C_NOPOS,
0565                FDL$C_NONE :            WRITE (FDL_DEST,'       none');
0566                FDL$C_IDX :             WRITE (FDL_DEST,'       indexed');
0567                FDL$C_REL :             WRITE (FDL_DEST,'       relative');
0568                FDL$C_SEQ :             WRITE (FDL_DEST,'       sequential');
0569                FDL$C_IF_IN :           WRITE (FDL_DEST,'       if_in_recovery_unit');
0570                FDL$C_NEC :             WRITE (FDL_DEST,'       necessary_to_write');
0571                FDL$C_NEVER :           WRITE (FDL_DEST,'       never_RU_journal');
0572                FDL$C_CR :              WRITE (FDL_DEST,'       carriage_return');
0573                FDL$C_FTN :             WRITE (FDL_DEST,'       FORTRAN');
0574                FDL$C_PRINT :           WRITE (FDL_DEST,'       print');
0575                FDL$C_FIX :             WRITE (FDL_DEST,'       fixed');
0576                FDL$C_STM :             WRITE (FDL_DEST,'       stream');
0577                FDL$C_STMCR :           WRITE (FDL_DEST,'       stream_CR');
0578                FDL$C_STMLF :           WRITE (FDL_DEST,'       stream_LF');
0579                FDL$C_UDF :             WRITE (FDL_DEST,'       undefined');
0580                FDL$C_VAR :             WRITE (FDL_DEST,'       variable');
0581                FDL$C_VFC :             WRITE (FDL_DEST,'       VFC');
0582                FDL$C_BN2 :             WRITE (FDL_DEST,'       bin2');
0583                FDL$C_BN4 :             WRITE (FDL_DEST,'       bin4');
0584                FDL$C_BN8 :             WRITE (FDL_DEST,'       bin8');
0585                FDL$C_PAC :             WRITE (FDL_DEST,'       decimal');
0586                FDL$C_IN2 :             WRITE (FDL_DEST,'       int2');
0587                FDL$C_IN4 :             WRITE (FDL_DEST,'       int4');
0588                FDL$C_IN8 :             WRITE (FDL_DEST,'       int8');
0589                FDL$C_STG :             WRITE (FDL_DEST,'       string');
0590                FDL$C_IAS :             WRITE (FDL_DEST,'       IAS');
0591                FDL$C_RSTS :            WRITE (FDL_DEST,'       RSTS/E');
0592                FDL$C_M :               WRITE (FDL_DEST,'       RSX-11M');
0593                FDL$C_MPLUS :           WRITE (FDL_DEST,'       RSX-11M-PLUS');
0594                FDL$C_RT :              WRITE (FDL_DEST,'       RT-11');
0595                FDL$C_TRAX :            WRITE (FDL_DEST,'       TRAX-11');
0596                FDL$C_VMS :             WRITE (FDL_DEST,'       VAX/VMS');
0597
0598            OTHERWISE
0599
0600                { NULL-STATEMENT } ;
0601
0602            END;        { CASE }
0603
0604        END;    { SHOW_QUALIFIER }
```

```
0606            { ++
0607
0608            SHOW_CURRENT -- Display (or output) the current line_object.
0609
0610            This routine outputs the textual representation of the line_object pointed
0611            to by DEF_HEAD.
0612
0613            CALLING SEQUENCE:
0614
0615            SHOW_CURRENT (PLUS_VALUE);
0616
0617            INPUT PARAMETERS:
0618
0619            none
0620
0621            IMPLICIT INPUTS:
0622
0623            DEF_CURRENT
0624            DEST_IS_TERMINAL
0625            LINES_PER_PAGE
0626
0627            OUTPUT PARAMETERS:
0628
0629            none
0630
0631            IMPLICIT OUTPUTS:
0632
0633            FDL_DEST
0634
0635            ROUTINES CALLED:
0636
0637            SHOW_PRIMARY
0638            SHOW_SECONDARY
0639            SHOW_QUALIFIER
0640            CLEAR
0641            LIB$SIGNAL
0642
0643            ROUTINE VALUE:
0644
0645            none
0646
0647            SIGNALS:
0648
0649
0650            SIDE EFFECTS:
0651
0652            none
0653
0654            -- }
```

```
0656        PROCEDURE SHOW_CURRENT (PLUS_VALUE : BOOLEAN);
0657
0658        VAR
0659            RETLEN      : [VOLATILE]$WORD;
0660
0661            PROCEDURE SHOW_PROT (PROTECTION : CTRL_ARRAY; FIELD_OFFSET : INTEGER);
0662
0663            BEGIN
0664
0665                IF PROTECTION[FIELD_OFFSET+EDF$V_NOREAD] THEN
0666
0667                    WRITE (FDL_DEST,'R');
0668
0669                IF PROTECTION[FIELD_OFFSET+EDF$V_NOWRITE] THEN
0670
0671                    WRITE (FDL_DEST,'W');
0672
0673                IF PROTECTION[FIELD_OFFSET+EDF$V_NOEXE] THEN
0674
0675                    WRITE (FDL_DEST,'E');
0676
0677                IF PROTECTION[FIELD_OFFSET+EDF$V_NODEL] THEN
0678
0679                    WRITE (FDL_DEST,'D');
0680
0681            END;        { SHOW_PROT }
0682
0683        BEGIN
0684
0685            { +
0686            If it's going to the terminal, shift it.
0687            - }
0688            IF DEST_IS_TERMINAL THEN
0689
0690                WRITE (FDL_DEST,SHIFT);
0691
0692            { +
0693            Write to FDL_DEST, according to the type of object it is.
0694            - }
0695            CASE DEF_CURRENT^.OBJECT_TYPE OF
0696
0697                PRI :
0698
0699                WITH DEF_CURRENT^ DO
0700
0701                BEGIN
0702
0703                    { +
0704                    Output the primary.
0705                    - }
0706                    SHOW_PRIMARY;
0707
0708                    { +
0709                    Show only the primary if wanted.
0710                    - }
0711                    IF PLUS_VALUE THEN
0712
```

```
0713                        { +
0714                        If it has an end-of-line comment on it, output that also.
0715                        - }
0716                        IF (COMMENT.DSC$W_LENGTH > 0) THEN
0717
0718                            WRITE (FDL_DEST,'    ',COMMENT.DSC$A_POINTER^:
0719                            COMMENT.DSC$W_LENGTH);
0720
0721                    { +
0722                    This actually does the QIO.
0723                    - }
0724                    WRITELN (FDL_DEST);
0725
0726                END;    { PRI }
0727
0728                SEC :
0729
0730                WITH DEF_CURRENT^ DO
0731
0732                BEGIN
0733
0734                    { +
0735                    Output the secondary.
0736                    - }
0737                    SHOW_SECONDARY;
0738
0739                    { +
0740                    Don't show the value unless wanted.
0741                    - }
0742                    IF PLUS_VALUE THEN
0743
0744                    BEGIN
0745
0746                        { +
0747                        If it's a string valued secondary, put the string out.
0748                        - }
0749                        IF SEC_TYPE[SECONDARY].STR THEN
0750
0751                        BEGIN
0752
0753                            CHECK_QUOTES (STRING);
0754
0755                            IF (STRING.DSC$W_LENGTH > 0) THEN
0756
0757                                WRITE (FDL_DEST,'       ',STRING.DSC$A_POINTER^:
0758                                STRING.DSC$W_LENGTH);
0759
0760                        END;
0761
0762                        { +
0763                        If it's a qualifier valued secondary, put the qualifier out.
0764                        - }
0765                        IF SEC_TYPE[SECONDARY].QUAL THEN
0766
0767                            SHOW_QUALIFIER;
0768
0769                        { +
```

```
0770                    If it's a number valued secondary, put the number out.
0771                    - }
0772                    IF SEC_TYPE[SECONDARY].NUM THEN
0773
0774                        { +
0775                        Show the number.
0776                        - }
0777                        WRITE (FDL_DEST,'   ',NUMBER:NUM_LEN(NUMBER));
0778
0779                    { +
0780                    If it's a boolean valued secondary, put YES or NO out.
0781                    - }
0782                    IF SEC_TYPE[SECONDARY].SW THEN
0783
0784                        IF SWITCH THEN
0785
0786                            WRITE (FDL_DEST,'       yes')
0787
0788                        ELSE
0789
0790                            WRITE (FDL_DEST,'       no');
0791
0792                    IF (
0793                    (SECONDARY = NULL_VALUE)
0794                    OR
0795                    (SECONDARY = MT_PROTECTION)
0796                    ) THEN
0797
0798                    BEGIN
0799
0800                        IF (
0801                        (NUMBER < %X20)     { SPACE }
0802                        OR
0803                        (NUMBER > %X7E)     { ~ }
0804                        ) THEN
0805
0806                            WRITE (FDL_DEST,'       ',NUMBER:NUM_LEN(NUMBER))
0807
0808                        ELSE
0809
0810                            WRITE (FDL_DEST,'       ''',CHR(NUMBER),'''');
0811
0812                    END;    { IF SECONDARY = NULL_VALUE OR MT_PROTECTION }
0813
0814                    { +
0815                    Area Position is a special case.
0816                    - }
0817                    IF SECONDARY = POSITION$ THEN
0818
0819                    BEGIN
0820
0821                        SHOW_QUALIFIER;
0822
0823                        IF (
0824                        QUALIFIER IN [ FDL$C_CYLPOS, FDL$C_LOGPOS, FDL$C_VIRPOS, FDL$C_CLUSPOS ]
0825                        ) THEN
0826
```

```
0827                    { +
0828                    Show the number.
0829                    - }
0830                    WRITE (FDL_DEST,'       ',NUMBER:NUM_LEN(NUMBER))
0831
0832                ELSE
0833
0834                BEGIN
0835
0836                    CASE QUALIFIER OF
0837
0838                        FDL$C_FIDPOS :
0839
0840                        BEGIN
0841
0842                            WRITE (FDL_DEST,
0843                            '       (',FID1:NUM_LEN(FID1),',',
0844                            FID2:NUM_LEN(FID2),',',
0845                            FID3:NUM_LEN(FID3),')');
0846
0847                        END;        { FILE_ID }
0848
0849                        FDL$C_FNMPOS :
0850
0851                        BEGIN
0852
0853                            CHECK_QUOTES (STRING);
0854
0855                            IF (STRING.DSC$W_LENGTH > 0) THEN
0856
0857                                WRITE (FDL_DEST,'   ',STRING.DSC$A_POINTER^:
0858                                STRING.DSC$W_LENGTH);
0859
0860                        END;        { FILE_NAME }
0861
0862                    OTHERWISE
0863
0864                        { NULL-STATEMENT } ;
0865
0866                    END;    { CASE }
0867
0868                END;        { IF FALSE QUALIFIER IN [ ... ] }
0869
0870            END;    { IF SECONDARY = POSITION$ }
0871
0872            { +
0873            OWNER is a special case.
0874            Use the special FAO directive to format the UIC.
0875            - }
0876            IF (SECONDARY = OWNER) THEN
0877
0878            BEGIN
0879
0880                TEMP_INT2 := OWNER_UIC;
0881                $FAO[ ('! !%I',RETLEN,TEMP_STRING255,TEMP_INT2);
0882                WRITE (FDL_DEST,TEMP_STRING255:RETLEN);
0883
```

```
0884                    END;
0885
0886                    { +
0887                    PROTECTION is also a special case.
0888                    - }
0889                    IF (SECONDARY = PROTECTION) THEN
0890
0891                    BEGIN
0892
0893                        WRITE (FDL_DEST,'   (system:');
0894                        SHOW_PROT (PROT_MASK,EDF$V_SYS);
0895                        WRITE (FDL_DEST,', owner:');
0896                        SHOW_PROT (PROT_MASK,EDF$V_OWN);
0897                        WRITE (FDL_DEST,', group:');
0898                        SHOW_PROT (PROT_MASK,EDF$V_GRP);
0899                        WRITE (FDL_DEST,', world:');
0900                        SHOW_PROT (PROT_MASK,EDF$V_WLD);
0901                        WRITE (FDL_DEST,')');
0902
0903                    END;    { IF TRUE SECONDARY = PROTECTION }
0904
0905                    { +
0906                    If it has an end-of-line comment, put that out too.
0907                    - }
0908                    IF (COMMENT.DSC$W_LENGTH > 0) THEN
0909
0910                        WRITE (FDL_DEST,'   ',COMMENT.DSC$A_POINTER^:
0911                        COMMENT.DSC$W_LENGTH);
0912
0913                        { +
0914                        This actually does the QIO.
0915                        - }
0916                        WRITELN (FDL_DEST);
0917
0918                    END;        { IF PLUS_VALUE }
0919
0920                END;    { SEC }
0921
0922                COM :
0923
0924                WITH DEF_CURRENT^ DO
0925
0926                BEGIN
0927
0928                    { +
0929                    This is a full-line comment. Just output it.
0930                    - }
0931                    IF (COMMENT.DSC$W_LENGTH > 0) THEN
0932
0933                        WRITELN (FDL_DEST,COMMENT.DSC$A_POINTER^:
0934                        COMMENT.DSC$W_LENGTH);
0935
0936                END;    { COM }
0937
0938            OTHERWISE
0939
0940                { NULL-STATEMENT } ;
```

```
0941
0942            END;        { CASE }
0943
0944            { +
0945            Keep track of the number of lines shown.
0946            - }
0947            LINES_SHOWN := LINES_SHOWN + 1;
0948
0949            IF DEST_IS_TERMINAL THEN
0950
0951            BEGIN
0952
0953                IF DEF_CURRENT^.FORE <> NIL THEN
0954
0955                    IF (
0956                    (((LINES_PER_PAGE - EDF$C_HEADROOM) - LINES_SHOWN) < 4)
0957                    AND
0958                    ((DEF_CURRENT^.PRIMARY <> DEF_CURRENT^.FORE^.PRIMARY)
0959                    OR
0960                    (DEF_CURRENT^.PRINUM <> DEF_CURRENT^.FORE^.PRINUM))
0961                    ) THEN
0962
0963                    BEGIN
0964
0965                        { +
0966                        We're about to output a short primary, reset and
0967                        clear the screen (after the user hits RETURN).
0968                        - }
0969                        LINES_SHOWN     := 0;
0970                        CLEAR (PAUSE);
0971
0972                    END;
0973
0974                IF (
0975                (LINES_SHOWN >= (LINES_PER_PAGE - EDF$C_HEADROOM))
0976                ) THEN
0977
0978                BEGIN
0979
0980                    { +
0981                    The counter tripped, reset and clear the screen
0982                    (after the user hits RETURN).
0983                    - }
0984                    LINES_SHOWN := 0;
0985                    CLEAR (PAUSE);
0986
0987                END;
0988
0989            END;        { IF DEST_IS_TERMINAL }
0990
0991        END;    { SHOW_CURRENT }
```

```
0993            { ++
0994
0995            GENERATE_FDL -- Routine to output the FDL definition.
0996
0997            This routine outputs the FDL definition.
0998
0999            CALLING SEQUENCE:
1000
1001            GENERATE_FDL;
1002
1003            INPUT PARAMETERS:
1004
1005            none
1006
1007            IMPLICIT INPUTS:
1008
1009            DEF_CURRENT
1010            ANSI_REVERSE
1011
1012            OUTPUT PARAMETERS:
1013
1014            none
1015
1016            IMPLICIT OUTPUTS:
1017
1018            FDL_DEST
1019            LINES_SHOWN
1020
1021            ROUTINES CALLED:
1022
1023            CLEAR
1024            SHOW_CURRENT
1025            INCR_CURRENT
1026            LIB$SIGNAL
1027
1028            ROUTINE VALUE:
1029
1030            none
1031
1032            SIGNALS:
1033
1034
1035            SIDE EFFECTS:
1036
1037            none
1038
1039            -- }
```

```
1041            PROCEDURE GENERATE_FDL;
1042
1043            VAR
1044                PREV_PRIMARY        : PRIMARY_TYPE;
1045                PREV_PRINUM         : INTEGER;
1046
1047            BEGIN
1048
1049                { +
1050                Do the Primaries as stored in the Definition Linked List.
1051                - }
1052                DEF_CURRENT := DEF_HEAD;
1053
1054                { +
1055                Setup to keep track of new primaries.
1056                - }
1057                PREV_PRINUM := -1;
1058
1059                { +
1060                Initialize the line counter (incremented in SHOW_CURRENT).
1061                - }
1062                LINES_SHOWN := 0;
1063
1064                { +
1065                Do it, if there is something to show.
1066                - }
1067                IF DEF_CURRENT <> NIL THEN
1068
1069                BEGIN
1070
1071                    { +
1072                    List isn't empty, cycle through the line_objects and show them.
1073                    At least until the end of the list, or until the user types ^Z.
1074                    - }
1075
1076                    REPEAT
1077
1078                        { +
1079                        Produce the textual version of the Definition Linked List.
1080                        Precede the 1st occurance of a primary with 2 blank lines.
1081                        - }
1082                        WITH DEF_CURRENT^ DO
1083
1084                        BEGIN
1085
1086                            IF PREV_PRINUM < 0 THEN
1087
1088                            BEGIN
1089
1090                                { +
1091                                Initial primary, don't skip lines here, just setup.
1092                                - }
1093                                PREV_PRIMARY        := PRIMARY;
1094                                PREV_PRINUM         := PRINUM;
1095
1096                            END
1097
```

```
1098                    ELSE
1099
1100                    BEGIN
1101
1102                        { +
1103                        Skip a line if this is a new primary.
1104                        - }
1105                        IF NOT (
1106                        (PREV_PRIMARY = PRIMARY) AND (PREV_PRINUM = PRINUM)
1107                        ) THEN
1108
1109                        BEGIN;
1110
1111                            PREV_PRIMARY    := PRIMARY;
1112                            PREV_PRINUM     := PRINUM;
1113
1114                            { +
1115                            This is a new primary, put out a blank line.
1116                            - }
1117                            WRITELN (FDL_DEST);
1118                            LINES_SHOWN     := LINES_SHOWN + 1;
1119
1120                        END;
1121
1122                    END;    { IF FALSE PREV_PRINUM >= 0 }
1123
1124                END;        { WITH DEF_CURRENT^ }
1125
1126                SHOW_CURRENT (TRUE);
1127                INCR_CURRENT;
1128
1129            UNTIL DEF_CURRENT = NIL;
1130
1131        END;        { IF FALSE DEF_CURRENT = NIL }
1132
1133    END;    { GENERATE_FDL }
```

```
1135            { ++
1136
1137            VIEW_DEF -- Routine to show the user the Definition Linked List.
1138
1139            This routine will display the definition on the user's terminal.
1140
1141            CALLING SEQUENCE:
1142
1143            VIEW_DEF;
1144
1145            INPUT PARAMETERS:
1146
1147            none
1148
1149            IMPLICIT INPUTS:
1150
1151            SYS$OUTPUT_NAME
1152            DEF_CURRENT
1153            CONTROL_ZEE_TYPED
1154
1155            OUTPUT PARAMETERS:
1156
1157            none
1158
1159            IMPLICIT OUTPUTS:
1160
1161            SYS$INPUT: the terminal
1162            DEST_IS_TERMINAL
1163
1164            ROUTINES CALLED:
1165
1166            CLEAR
1167            GENERATE_FDL
1168
1169            ROUTINE VALUE:
1170
1171            none
1172
1173            SIGNALS:
1174
1175            none
1176
1177            SIDE EFFECTS:
1178
1179            none
1180
1181            -- }
```

```
1183            PROCEDURE VIEW_DEF;
1184
1185            BEGIN
1186
1187                { +
1188                Erase the user's screen.
1189                - }
1190                CLEAR (SCREEN);
1191
1192                { +
1193                'Open' his terminal and initialize it.
1194                Close it first to avoid conflicts.
1195                - }
1196                DEST_IS_TERMINAL    := TRUE;
1197
1198                CLOSE       (FDL_DEST,ERROR := CONTINUE);
1199                OPEN        (FDL_DEST,SYS$OUTPUT_NAME,NEW,RECORD_LENGTH := 252);
1200                REWRITE     (FDL_DEST);
1201
1202                { +
1203                Put the current definition out to the terminal.
1204                - }
1205                GENERATE_FDL;
1206
1207                { +
1208                We're done, close it off.
1209                - }
1210                CLOSE       (FDL_DEST);
1211
1212                { +
1213                Don't clear if the user hit ^Z, or if SHOW_CURRENT had just done a PAUSE.
1214                - }
1215                IF (NOT CONTROL_ZEE_TYPED) AND (LINES_SHOWN <> 0) THEN
1216
1217                BEGIN
1218
1219                    WRITELN;
1220                    CLEAR (PAUSE);
1221
1222                END;
1223
1224            END;    { VIEW_DEF }
```

```
1226            { ++
1227
1228            SHOW_PRIMARY_SECTION -- Display the whole primary section.
1229
1230            This routine outputs the selected primary section to the screen.
1231
1232            CALLING SEQUENCE:
1233
1234            SHOW_PRIMARY_SECTION (TEST);
1235
1236            INPUT PARAMETERS:
1237
1238            TEST
1239
1240            IMPLICIT INPUTS:
1241
1242            FDL_DEST
1243            CONTROL_ZEE_TYPED
1244            DEF_CURRENT
1245            DEF_HEAD
1246
1247            OUTPUT PARAMETERS:
1248
1249            none
1250
1251            IMPLICIT OUTPUTS:
1252
1253            SYS$OUTPUT:
1254            DEF_CURRENT
1255
1256            ROUTINES CALLED:
1257
1258            CLEAR
1259            INCR_CURRENT
1260            SECTION_MATCH
1261
1262            ROUTINE VALUE:
1263
1264            none
1265
1266            SIGNALS:
1267
1268            none
1269
1270            SIDE EFFECTS:
1271
1272            none
1273
1274            -- }
```

```
1276        [GLOBAL] PROCEDURE SHOW_PRIMARY_SECTION (TEST : LINE_OBJECT);
1277
1278        VAR
1279            AT_PRIMARY  : BOOLEAN;
1280
1281        BEGIN
1282
1283            { +
1284            Show him that whole primary section.
1285            - }
1286            LINES_SHOWN         := 0;
1287
1288            { +
1289            Step through the whole list.
1290            - }
1291            DEF_CURRENT         := DEF_HEAD;
1292            AT_PRIMARY          := FALSE;
1293
1294            REPEAT
1295
1296                { +
1297                If DEF_CURRENT points to a line_object in that primary, show it.
1298                - }
1299                IF CURRENT_EQ_TEST (TEST,FALSE) THEN
1300
1301                BEGIN
1302
1303                    SHOW_CURRENT (TRUE);
1304                    AT_PRIMARY  := TRUE;
1305
1306                END
1307                ELSE
1308
1309                BEGIN
1310
1311                    IF AT_PRIMARY THEN
1312
1313                        DEF_CURRENT     := NIL;
1314
1315                END;
1316
1317                { +
1318                Try the next.
1319                - }
1320                INCR_CURRENT;
1321
1322            UNTIL DEF_CURRENT = NIL;
1323
1324            WRITELN (FDL_DEST);
1325
1326        END;    { SHOW_PRIMARY_SECTION }
1327
```

```
1329            { ++
1330
1331            SHOW_ALL_PRIMARIES -- Display the existing primary attributes.
1332
1333            This routine outputs the existing primaries on the screen.
1334
1335            CALLING SEQUENCE:
1336
1337            SHOW_ALL_PRIMARIES;
1338
1339            INPUT PARAMETERS:
1340
1341            none
1342
1343            IMPLICIT INPUTS:
1344
1345            FDL_DEST
1346            CONTROL_ZEE_TYPED
1347            DEF_CURRENT
1348            DEF_HEAD
1349
1350            OUTPUT PARAMETERS:
1351
1352            none
1353
1354            IMPLICIT OUTPUTS:
1355
1356            SYS$OUTPUT:
1357            DEF_CURRENT
1358
1359            ROUTINES CALLED:
1360
1361            CLEAR
1362            INCR_CURRENT
1363            SECTION_MATCH
1364
1365            ROUTINE VALUE:
1366
1367            none
1368
1369            SIGNALS:
1370
1371            none
1372
1373            SIDE EFFECTS:
1374
1375            none
1376
1377            -- }
```

```
1379            [GLOBAL] PROCEDURE SHOW_ALL_PRIMARIES;
1380
1381            BEGIN
1382
1383                { +
1384                Show him all the primary attributes.
1385                - }
1386                LINES_SHOWN         := 0;
1387
1388                { +
1389                Step through the whole list.
1390                - }
1391                DEF_CURRENT         := DEF_HEAD;
1392
1393                REPEAT
1394
1395                    { +
1396                    If DEF_CURRENT points to a line_object that is a primary, show it.
1397                    But not the Ident.
1398                    - }
1399                    IF (
1400                    (DEF_CURRENT^.OBJECT_TYPE = PRI)
1401                    AND
1402                    (DEF_CURRENT^.PRIMARY <> IDENT)
1403                    ) THEN
1404
1405                        SHOW_CURRENT (TRUE);
1406
1407                    { +
1408                    Try the next.
1409                    - }
1410                    INCR_CURRENT;
1411
1412                UNTIL DEF_CURRENT = NIL;
1413
1414                WRITELN (FDL_DEST);
1415
1416            END;    { SHOW_ALL_PRIMARIES }
```

```
1418            { ++
1419
1420            SHOW_CUR_PRI_SEC -- Display the line_object pointed to by def_current.
1421
1422            This routine outputs the def_current line_object.
1423
1424            CALLING SEQUENCE:
1425
1426            SHOW_CUR_PRI_SEC;
1427
1428            INPUT PARAMETERS:
1429
1430            none
1431
1432            IMPLICIT INPUTS:
1433
1434            FDL_DEST
1435            CONTROL_ZEE_TYPED
1436            DEF_CURRENT
1437            DEF_HEAD
1438
1439            OUTPUT PARAMETERS:
1440
1441            none
1442
1443            IMPLICIT OUTPUTS:
1444
1445            SYS$OUTPUT:
1446            DEF_CURRENT
1447
1448            ROUTINES CALLED:
1449
1450            CLEAR
1451            INCR_CURRENT
1452            SECTION_MATCH
1453
1454            ROUTINE VALUE:
1455
1456            none
1457
1458            SIGNALS:
1459
1460            none
1461
1462            SIDE EFFECTS:
1463
1464            none
1465
1466            -- }
```

```
1468            [GLOBAL] PROCEDURE SHOW_CUR_PRI_SEC (PLUS_VALUE : BOOLEAN);
1469
1470            VAR
1471                SAVE_OBJECT_TYPE            : LINE_OBJECT_TYPE;
1472
1473            BEGIN
1474
1475                LINES_SHOWN                 := 0;
1476
1477                { +
1478                Now display the 2 versions of the line_object.
1479                All this only works because the primary that a line_object is in
1480                is stored redundantly in all the subsequent secondaries.
1481                - }
1482                SAVE_OBJECT_TYPE            := DEF_CURRENT^.OBJECT_TYPE;
1483
1484                DEF_CURRENT^.OBJECT_TYPE    := PRI;
1485                SHOW_CURRENT (PLUS_VALUE);
1486
1487                IF DEF_CURRENT^.PRIMARY <> TITLE THEN
1488
1489                BEGIN
1490
1491                    DEF_CURRENT^.OBJECT_TYPE := SEC;
1492                    SHOW_CURRENT (PLUS_VALUE);
1493
1494                END;
1495
1496                DEF_CURRENT^.OBJECT_TYPE    := SAVE_OBJECT_TYPE;
1497
1498            END;    { SHOW_CUR_PRI_SEC }
1499
1500            END.
1501                { End of file: SRC$:EDFSHOW.PAS }
```

```
                                                                                   .TITLE  EDFSHOW
                                                                                   .IDENT  \V04-000\

                                                                     00000         .PSECT  $CODE,PIC,CON,REL,LCL,SHR,EXE,RD,NOWRT,2

        0000006B  00000064  00000055  00000010  00000048   00000 C.AAA:  .LONG   72,16,85,100,107,111,128,144,149,157,162,-
        0000009D  00000095  00000090  00000080  0000006F   00014                 168,174,182,186,194,202,209
        000000BA  000000B6  000000AE  000000A8  000000A2   00028
                            000000D1  000000CA  000000C2   0003C
   45 50 59 54 5F 59 52 41 4D 49 52 50 0C                  00048         .ASCII  <12>\PRIMARY_TYPE\
59 52 41 4D 49 52 50 5F 59 4D 4D 55 44 0E                  00055         .ASCII  <14>\DUMMY_PRIMARY$\
                                       24                  00063
                     53 53 45 43 43 41 06                  00064         .ASCII  <6>\ACCESS\
                                 4C 43 41 03               0006B         .ASCII  <3>\ACL\
41 5F 46 4F 5F 53 49 53 59 4C 41 4E 41 10                  0006F         .ASCII  <16>\ANALYSIS_OF_AREA\
                                    41 45 52               0007D
4B 5F 46 4F 5F 53 49 53 59 4C 41 4E 41 0F                  00080         .ASCII  <15>\ANALYSIS_OF_KEY\
                                       59 45               0008E
                              41 45 52 41 04               00090         .ASCII  <4>\AREA\
                     54 43 45 4E 4E 4F 43 07               00095         .ASCII  <7>\CONNECT\
                              45 54 41 44 04               0009D         .ASCII  <4>\DATE\
                           24 45 4C 49 46 05               000A2         .ASCII  <5>\FILES\
                           54 4E 45 44 49 05               000A8         .ASCII  <5>\IDENT\
                     4C 41 4E 52 55 4F 4A 07               000AE         .ASCII  <7>\JOURNAL\
                                 59 45 4B 03               000B6         .ASCII  <3>\KEY\
                     24 44 52 4F 43 45 52 07               000BA         .ASCII  <7>\RECORDS$\
                     47 4E 49 52 41 48 53 07               000C2         .ASCII  <7>\SHARING\
                        4D 45 54 53 59 53 06               000CA         .ASCII  <6>\SYSTEM\
                        00 45 4C 54 49 54 05               000D1         .ASCII  <5>\TITLE\<0>
                                       08 38               000D8 C.AAB:  .BYTE   ^X38,8
                                       82 00               000DA C.AAC:  .BYTE   0,^X82
                  00000000  00000000  00000000  00000000   000DC C.AAD:  .LONG   0,0,0,0
                                           00 00 E0        000EC         .BYTE   ^XE0,0,0
                                                           000EF         .BLKB   1
        0000028E  00000284  00000273  00000097  00000264   000F0 C.AAE:  .LONG   612,151,627,644,654,662,667,672,683,693,-
        000002B5  000002AB  000002A0  0000029B  00000296   00104                 701,707,723,734,756,781,799,819,829,835,-
        000002F4  000002DE  000002D3  000002C3  000002BD   00118                 855,874,886,907,927,944,962,978,993,1013,-
        00000343  0000033D  00000333  0000031F  0000030D   0012C                 1025,1046,1059,1071,1090,1101,1111,1119,-
        0000039F  0000038B  00000376  0000036A  00000357   00140                 1132,1141,1153,1161,1173,1186,1198,1215,-
        000003F5  000003E1  000003D2  000003C2  000003B0   00154                 1233,1250,1260,1272,1285,1299,1316,1333,-
        00000442  0000042F  00000423  00000416  00000401   00168                 1351,1358,1377,1388,1404,1419,1434,1450,-
        00000475  0000046C  0000045F  00000457  0000044D   0017C                 1470,1486,1496,1516,1531,1548,1558,1574,-
        000004AE  000004A2  00000495  00000489  00000481   00190                 1587,1595,1605,1617,1627,1638,1658,1670,-
        000004F8  000004EC  000004E2  000004D1  000004BF   001A4                 1683,1692,1703,1713,1726,1741,1757,1773,-
        00000547  00000535  00000524  00000513  00000505   001B8                 1789,1799,1819,1833,1853,1864,1878,1893,-
        0000058B  0000057C  0000056C  00000561  0000054E   001CC                 1909,1927,1944,1949,1958,1978,1996,2009,-
        000005D8  000005CE  000005BE  000005AA  0000059A   001E0                 2015,2030,2041,2052,2061,2077,2093,2103,-
        00000626  00000616  0000060C  000005FB  000005EC   001F4                 2113,2131,2146,2158,2170,2182,2193,2205,-
        0000065B  00000651  00000645  0000063B  00000633   00208                 2216,2229,2241,2255,2263,2273,2283,2304,-
        0000069C  00000693  00000686  0000067A  00000666   0021C                 2328,2339,2350,2368,2379,2397,2403,2412,-
        000006DD  000006CD  000006BE  000006B1  000006A7   00230                 2423,2432,2443,2456
        00000729  0000071B  00000707  000006FD  000006ED   00244
        00000775  00000765  00000756  00000748  0000073D   00258
        000007BA  000007A6  0000079D  00000798  00000787   0026C
        000007F9  000007EE  000007DF  000007D9  000007CC   00280
        00000837  0000082D  0000081D  0000080D  00000804   00294
        0000087A  0000086E  00000862  00000853  00000841   002A8
```

```
                000008B5  000008A8  0000089D  00000891  00000886      002BC
                000008EB  000008E1  000008D7  000008CF  000008C1      002D0
                00000940  0000092E  00000923  00000918  00000900      002E4
                00000977  0000096C  00000963  0000095D  0000094B      002F8
                                              0000098B  00000980      0030C
                000009D7  000009D0  000009BD  000009AC  000009A1      00314          .LONG   2465,2476,2493,2512,2519,2524,2531,2535,-
                000009FC  000009F3  000009E7  000009E3  000009DC      00328                  2547,2556,2560,2567,2582,2589,2596
                00000A24  00000A1D  00000A16  00000A07  00000A00      0033C
 50  59  54  5F  59  52  41  44  4E  4F  43  45  53  0E              00350          .ASCII  <14>\SECONDARY_TYPE\
                                                      45              0035E
 41  44  4E  4F  43  45  53  5F  59  4D  4D  55  44  10              0035F          .ASCII  <16>\DUMMY_SECONDARY$\
                                                  24  59  52          0036D
                      24  4F  49  5F  4B  43  4F  4C  42  09          00370          .ASCII  <9>\BLOCK_IO$\
                          24  45  54  45  4C  45  44  07              0037A          .ASCII  <7>\DELETE$\
                              24  54  45  47  04                      00382          .ASCII  <4>\GET$\
                              24  54  55  50  04                      00387          .ASCII  <4>\PUT$\
                  24  4F  49  5F  44  52  4F  43  45  52  0A          0038C          .ASCII  <10>\RECORD_IO$\
                      24  45  54  41  43  4E  55  52  54  09          00397          .ASCII  <9>\TRUNCATE$\
                          24  45  54  41  44  50  55  07              003A1          .ASCII  <7>\UPDATE$\
                              59  52  54  4E  45  05                  003A9          .ASCII  <5>\ENTRY\
 41  50  53  5F  44  45  4D  49  41  4C  43  45  52  0F              003AF          .ASCII  <15>\RECLAIMED_SPACE\
                                                      45  43          003BD
                  24  4C  4C  49  46  5F  41  54  41  44  0A          003BF          .ASCII  <10>\DATA_FILL$\
 50  4D  4F  43  5F  59  45  4B  5F  41  54  41  44  15              003CA          .ASCII  <21>\DATA_KEY_COMPRESSION$\
                          24  4E  4F  49  53  53  45  52              003D8
 43  5F  44  52  4F  43  45  52  5F  41  54  41  44  18              003E0          .ASCII  <24>\DATA_RECORD_COMPRESSION$\
                  24  4E  4F  49  53  53  45  52  50  4D  4F          003EE
 43  5F  44  52  4F  43  45  52  5F  41      54  41  44  11          003F9          .ASCII  <17>\DATA_RECORD_COUNT\
                                          54  4E  55  4F              00407
 43  4F  5F  45  43  41  50  53  5F  41  54  41  44  13              0040B          .ASCII  <19>\DATA_SPACE_OCCUPIED\
                                      44  45  49  50  55  43          00419
                      53  4E  4F  49  54  45  4C  45  44  09          0041F          .ASCII  <9>\DELETIONS\
                                  48  54  50  45  44  05              00429          .ASCII  <5>\DEPTH\
 45  50  5F  53  45  54  41  43  49  4C  50  55  44  13              0042F          .ASCII  <19>\DUPLICATES_PER_SIDR\
                                  52  44  49  53  5F  52              0043D
 53  45  52  50  4D  4F  43  5F  58  45  44  4E  49  12              00443          .ASCII  <18>\INDEX_COMPRESSION$\
                                      24  4E  4F  49  53              00451
                  24  4C  4C  49  46  5F  58  45  44  4E  49  0C      00456          .ASCII  <11>\INDEX_FILL$\
 4F  5F  45  43  41  50  53  5F  58  45  44  4E  49  14              00462          .ASCII  <20>\INDEX_SPACE_OCCUPIED\
                                  44  45  49  50  55  43  43          00470
 44  52  4F  43  45  52  5F  31  4C  45  56  45  4C  13              00477          .ASCII  <19>\LEVEL1_RECORD_COUNT\
                                  54  4E  55  4F  43  5F              00485
 4E  45  4C  5F  41  54  41  44  5F  4E  41  45  4D  10              0048B          .ASCII  <16>\MEAN_DATA_LENGTH\
                                              48  54  47          00499
 45  4C  5F  58  45  44  4E  49  5F  4E  41  45  4D  11              0049C          .ASCII  <17>\MEAN_INDEX_LENGTH\
                                          48  54  47  4E          004AA
 53  53  45  43  43  41  5F  4D  4F  44  4E  41  52  0F              004AE          .ASCII  <15>\RANDOM_ACCESSES\
                                                  53  45          004BC
 54  52  45  53  4E  49  5F  4D  4F  44  4E  41  52  0E              004BE          .ASCII  <14>\RANDOM_INSERTS\
                                                      53          004CC
 43  41  5F  4C  41  49  54  4E  45  55  51  45  53  13              004CD          .ASCII  <19>\SEQUENTIAL_ACCESSES\
                                  53  45  53  53  45  43          004DB
                  24  4E  4F  49  54  41  43  4F  4C  4C  41  0B      004E1          .ASCII  <11>\ALLOCATION$\
 54  4E  4F  43  5F  59  52  54  5F  54  53  45  42  14              004ED          .ASCII  <20>\BEST_TRY_CONTIGUOUS$\
                              24  53  55  4F  55  47  49          004FB
      24  45  5A  49  53  5F  54  45  4B  43  55  42  0C      00502          .ASCII  <12>\BUCKET_SIZE$\
          24  53  55  4F  55  47  49  54  4E  4F  43  0B      0050F          .ASCII  <11>\CONTIGUOUS$\
```

```
4F  49  54  49  53  4F  50  5F  54  43  41  58  45  12     0051B    .ASCII  <18>\EXACT_POSITIONING$\
                                    24  47  4E  49  4E     00529
            24  4E  4F  49  53  4E  45  54  58  45  0A     0052E    .ASCII  <10>\EXTENSION$\
                24  4E  4F  49  54  49  53  4F  50  09     00539    .ASCII  <9>\POSITION$\
                        24  45  4D  55  4C  4F  56  07     00543    .ASCII  <7>\VOLUME$\
    53  55  4F  4E  4F  52  48  43  4E  59  53  41  0C     0054B    .ASCII  <12>\ASYNCHRONOUS\
                    4F  49  5F  4B  43  4F  4C  42  08     00558    .ASCII  <8>\BLOCK_IO\
        45  44  4F  43  5F  54  45  4B  43  55  42  0B     00561    .ASCII  <11>\BUCKET_CODE\
                        54  58  45  54  4E  4F  43  07     0056D    .ASCII  <7>\CONTEXT\
        45  4C  49  46  5F  46  4F  5F  44  4E  45  0B     00575    .ASCII  <11>\END_OF_FILE\
    53  54  45  4B  43  55  42  5F  4C  4C  49  46  0C     00581    .ASCII  <12>\FILL_BUCKETS\
        45  54  45  4C  45  44  5F  54  53  41  46  0B     0058E    .ASCII  <11>\FAST_DELETE\
45  52  45  46  45  52  5F  46  4F  5F  59  45  4B  10     0059A    .ASCII  <16>\KEY_OF_REFERENCE\
                                            45  43  4E     005A8
45  5F  52  45  54  41  45  52  47  5F  59  45  4B  11     005AB    .ASCII  <17>\KEY_GREATER_EQUAL\
                                        4C  41  55  51     005B9
54  5F  52  45  54  41  45  52  47  5F  59  45  4B  10     005BD    .ASCII  <16>\KEY_GREATER_THAN\
                                            4E  41  48     005CB
                54  49  4D  49  4C  5F  59  45  4B  09     005CE    .ASCII  <9>\KEY_LIMIT\
        45  44  4F  4D  5F  45  54  41  43  4F  4C  0B     005D8    .ASCII  <11>\LOCATE_MODE\
    44  41  45  52  5F  4E  4F  5F  4B  43  4F  4C  0C     005E4    .ASCII  <12>\LOCK_ON_READ\
45  54  49  52  57  5F  4E  4F  5F  4B  43  4F  4C  0D     005F1    .ASCII  <13>\LOCK_ON_WRITE\
4B  43  4F  4C  4E  55  5F  4C  41  55  4E  41  4D  10     005FF    .ASCII  <16>\MANUAL_UNLOCKING\
                                            47  4E  49     0060D
4F  43  5F  4B  43  4F  4C  42  49  54  4C  55  4D  10     00610    .ASCII  <16>\MULTIBLOCK_COUNT\
                                            54  4E  55     0061E
43  5F  52  45  46  46  55  42  49  54  4C  55  4D  11     00621    .ASCII  <17>\MULTIBUFFER_COUNT\
                                        54  4E  55  4F     0062F
                            4B  43  4F  4C  4F  4E  06     00633    .ASCII  <6>\NOLOCK\
52  5F  54  4E  45  54  53  49  58  45  4E  4F  4E  12     0063A    .ASCII  <18>\NONEXISTENT_RECORD\
                                    44  52  4F  43  45     00648
            44  41  45  48  41  5F  44  41  45  52  0A     0064D    .ASCII  <10>\READ_AHEAD\
45  4C  44  52  41  47  45  52  5F  44  41  45  52  0F     00658    .ASCII  <15>\READ_REGARDLESS\
                                                53  53     00666
4C  42  41  4E  45  5F  54  55  4F  45  4D  49  54  0E     00668    .ASCII  <14>\TIMEOUT_ENABLE\
                                                    45     00676
4F  49  52  45  50  5F  54  55  4F  45  4D  49  54  0E     00677    .ASCII  <14>\TIMEOUT_PERIOD\
                                                    44     00685
50  5F  4E  4F  5F  45  54  41  43  4E  55  52  54  0F     00686    .ASCII  <15>\TRUNCATE_ON_PUT\
                                                54  55     00694
4E  4F  43  5F  4C  45  43  4E  41  43  5F  54  54  13     00696    .ASCII  <19>\TT_CANCEL_CONTROL_O\
                                4F  5F  4C  4F  52  54     006A4
50  4E  49  5F  45  53  41  43  50  55  5F  54  54  0F     006AA    .ASCII  <15>\TT_UPCASE_INPUT\
                                                54  55     006B8
                54  50  4D  4F  52  50  5F  54  54  09     006BA    .ASCII  <9>\TT_PROMPT\
45  50  59  54  5F  45  47  52  55  50  5F  54  54  13     006C4    .ASCII  <19>\TT_PURGE_TYPE_AHEAD\
                                44  41  45  48  41  5F     006D2
48  43  45  4F  4E  5F  44  41  45  52  5F  54  54  0E     006D8    .ASCII  <14>\TT_READ_NOECHO\
                                                    4F     006E6
4C  49  46  4F  4E  5F  44  41  45  52  5F  54  54  10     006E7    .ASCII  <16>\TT_READ_NOFILTER\
                                            52  45  54     006F5
                46  49  5F  45  54  41  44  50  55  09     006F8    .ASCII  <9>\UPDATE_IF\
4F  43  45  52  5F  52  4F  46  5F  54  49  41  57  0F     00702    .ASCII  <15>\WAIT_FOR_RECORD\
                                                44  52     00710
    44  4E  49  48  45  42  5F  45  54  49  52  57  0C     00712    .ASCII  <12>\WRITE_BEHIND\
                        24  50  55  4B  43  41  42  07     0071F    .ASCII  <7>\BACKUP$\
                24  4E  4F  49  54  41  45  52  43  09     00727    .ASCII  <9>\CREATION$\
```

```
                24 4E 4F 49 54 41 52 49 50 58 45 0B  00731          .ASCII  <11>\EXPIRATION$\
                   24 4E 4F 49 53 49 56 45 52 09     0073D          .ASCII  <9>\REVISION$\
                4E 4F 49 54 41 43 4F 4C 4C 41 0A     00747          .ASCII  <10>\ALLOCATION\
54 4E 4F 43 5F 59 52 54 5F 54 53 45 42 13            00752          .ASCII  <19>\BEST_TRY_CONTIGUOUS\
                   53 55 4F 55 47 49                 00760
             45 5A 49 53 5F 54 45 4B 43 55 42 0B     00766          .ASCII  <11>\BUCKET_SIZE\
          45 5A 49 53 5F 52 45 54 53 55 4C 43 0C     00772          .ASCII  <12>\CLUSTER_SIZE\
                   24 54 58 45 54 4E 4F 43 08        0077F          .ASCII  <8>\CONTEXT$\
             53 55 4F 55 47 49 54 4E 4F 43 0A        00788          .ASCII  <10>\CONTIGUOUS\
                46 49 5F 45 54 41 45 52 43 09        00793          .ASCII  <9>\CREATE_IF\
          45 4D 41 4E 5F 54 4C 55 41 46 45 44 0C     0079D          .ASCII  <12>\DEFAULT_NAME\
54 49 52 57 5F 44 45 52 52 45 46 45 44 0E            007AA          .ASCII  <14>\DEFERRED_WRITE\
                                         45          007B8
4F 4C 43 5F 4E 4F 5F 45 54 45 4C 45 44 0F            007B9          .ASCII  <15>\DELETE_ON_CLOSE\
                                         45 53       007C7
54 4E 45 5F 59 52 4F 54 43 45 52 49 44 0F            007C9          .ASCII  <15>\DIRECTORY_ENTRY\
                                         59 52       007D7
45 4C 45 44 5F 4E 4F 5F 45 53 41 52 45 0F            007D9          .ASCII  <15>\ERASE_ON_DELETE\
                                         45 54       007E7
                4E 4F 49 53 4E 45 54 58 45 09        007E9          .ASCII  <9>\EXTENSION\
52 45 46 46 55 42 5F 4C 41 42 4F 4C 47 13            007F3          .ASCII  <19>\GLOBAL_BUFFER_COUNT\
                   54 4E 55 4F 43 5F                 00801
45 5A 49 53 5F 4B 43 4F 4C 42 5F 54 4D 0D            00807          .ASCII  <13>\MT_BLOCK_SIZE\
4F 50 5F 54 4E 45 52 52 55 43 5F 54 4D 13            00815          .ASCII  <19>\MT_CURRENT_POSITION\
                   4E 4F 49 54 49 53                 00823
             46 4F 45 5F 54 4F 4E 5F 54 4D 0A        00829          .ASCII  <10>\MT_NOT_EOF\
4E 4F 49 54 43 45 54 4F 52 50 5F 54 4D 0D            00834          .ASCII  <13>\MT_PROTECTION\
4E 49 57 45 52 5F 4E 45 50 4F 5F 54 4D 0E            00842          .ASCII  <14>\MT_OPEN_REWIND\
                                         44          00850
49 57 45 52 5F 45 53 4F 4C 43 5F 54 4D 0F            00851          .ASCII  <15>\MT_CLOSE_REWIND\
                                         44 4E       0085F
55 4E 5F 44 52 4F 43 45 52 5F 58 41 4D 11            00861          .ASCII  <17>\MAX_RECORD_NUMBER\
                                         52 45 42 4D 0086F
53 52 45 56 5F 45 5A 49 4D 49 58 41 4D 10            00873          .ASCII  <16>\MAXIMIZE_VERSION\
                                         4E 4F 49    00881
                                   45 4D 41 4E 04     00884          .ASCII  <4>\NAME\
                      50 55 4B 43 41 42 4F 4E 08     00889          .ASCII  <8>\NOBACKUP\
55 52 54 53 5F 45 4C 49 46 5F 4E 4F 4E 13            00892          .ASCII  <19>\NON_FILE_STRUCTURED\
                                   44 45 52 55 54 43 008A0
50 5F 45 4C 49 46 5F 54 55 50 54 55 4F 11            008A6          .ASCII  <17>\OUTPUT_FILE_PARSE\
                                         45 53 52 41 008B4
          4E 4F 49 54 41 5A 49 4E 41 47 52 4F 0C     008B8          .ASCII  <12>\ORGANIZATION\
                                   52 45 4E 57 4F 05  008C5          .ASCII  <5>\OWNER\
53 4F 4C 43 5F 4E 4F 5F 54 4E 49 52 50 0E            008CB          .ASCII  <14>\PRINT_ON_CLOSE\
                                         45          008D9
             4E 4F 49 54 43 45 54 4F 52 50 0A        008DA          .ASCII  <10>\PROTECTION\
             4B 43 45 48 43 5F 44 41 45 52 0A        008E5          .ASCII  <10>\READ_CHECK\
                4E 4F 49 53 49 56 45 52 08           008F0          .ASCII  <8>\REVISION\
4E 4F 5F 4C 41 49 54 4E 45 55 51 45 53 0F            008F9          .ASCII  <15>\SEQUENTIAL_ONLY\
                                         59 4C       00907
4F 4C 43 5F 4E 4F 5F 54 49 4D 42 55 53 0F            00909          .ASCII  <15>\SUBMIT_ON_CLOSE\
                                         45 53       00917
                45 44 45 53 52 45 50 55 53 09        00919          .ASCII  <9>\SUPERSEDE\
                59 52 41 52 4F 50 4D 45 54 09        00923          .ASCII  <9>\TEMPORARY\
43 5F 4E 4F 5F 45 54 41 43 4E 55 52 54 11            0092D          .ASCII  <17>\TRUNCATE_ON_CLOSE\
                                   45 53 4F 4C       0093B
45 50 4F 5F 45 4C 49 46 5F 52 45 53 55 0E            0093F          .ASCII  <14>\USER_FILE_OPEN\
```

```
                                                                 4E      0094D
                45  5A  49  53  5F  57  4F  44  4E  49  57      0B      0094E          .ASCII  <11>\WINDOW_SIZE\
                4B  43  45  48  43  5F  45  54  49  52  57      0B      0095A          .ASCII  <11>\WRITE_CHECK\
                45  47  41  4D  49  5F  52  45  54  46  41      0B      00966          .ASCII  <11>\AFTER_IMAGE\
                    45  4D  41  4E  5F  52  45  54  46  41      0A      00972          .ASCII  <10>\AFTER_NAME\
                4C  49  41  52  54  5F  54  49  44  55  41      0B      0097D          .ASCII  <11>\AUDIT_TRAIL\
                    45  4D  41  4E  5F  54  49  44  55  41      0A      00989          .ASCII  <10>\AUDIT_NAME\
            45  47  41  4D  49  5F  45  52  4F  46  45  42      0C      00994          .ASCII  <12>\BEFORE_IMAGE\
                45  4D  41  4E  5F  45  52  4F  46  45  42      0B      009A1          .ASCII  <11>\BEFORE_NAME\
    54  49  4E  55  5F  59  52  45  56  4F  43  45  52      0D      009AD          .ASCII  <13>\RECOVERY_UNIT\
                            53  45  47  4E  41  48  43      07      009BB          .ASCII  <7>\CHANGES\
                        41  45  52  41  5F  41  54  41  44      09      009C3          .ASCII  <9>\DATA_AREA\
                        4C  4C  49  46  5F  41  54  41  44      09      009CD          .ASCII  <9>\DATA_FILL\
    50  4D  4F  43  5F  59  45  4B  5F  41  54  41  44      14      009D7          .ASCII  <20>\DATA_KEY_COMPRESSION\
                                4E  4F  49  53  53  45  52      00      009E5
    43  5F  44  52  4F  43  45  52  5F  41  54  41  44      17      009EC          .ASCII  <23>\DATA_RECORD_COMPRESSION\
                        4E  4F  49  53  53  45  52  50  4D  4F      009FA
                    53  45  54  41  43  49  4C  50  55  44  0A      00A04          .ASCII  <10>\DUPLICATES\
                    41  45  52  41  5F  58  45  44  4E  49  0A      00A0F          .ASCII  <10>\INDEX_AREA\
    53  45  52  50  4D  4F  43  5F  58  45  44  4E  49      11      00A1A          .ASCII  <17>\INDEX_COMPRESSION\
                                    4E  4F  49  53      00      00A28
                    4C  4C  49  46  5F  58  45  44  4E  49  0A      00A2C          .ASCII  <10>\INDEX_FILL\
    5F  58  45  44  4E  49  5F  31  4C  45  56  45  4C      11      00A37          .ASCII  <17>\LEVEL1_INDEX_AREA\
                                        41  45  52  41      00      00A45
                                24  45  4D  41  4E      05      00A49          .ASCII  <5>\NAME$\
                        59  45  4B  5F  4C  4C  55  4E      08      00A4F          .ASCII  <8>\NULL_KEY\
                    45  55  4C  41  56  5F  4C  4C  55  4E  0A      00A58          .ASCII  <10>\NULL_VALUE\
                        45  55  47  4F  4C  4F  52  50      08      00A63          .ASCII  <8>\PROLOGUE\
                    48  54  47  4E  45  4C  5F  47  45  53  0A      00A6C          .ASCII  <10>\SEG_LENGTH\
                4E  4F  49  54  49  53  4F  50  5F  47  45  53  0C      00A77          .ASCII  <12>\SEG_POSITION\
                        45  50  59  54  5F  47  45  53      08      00A84          .ASCII  <8>\SEG_TYPE\
                    4E  41  50  53  5F  4B  43  4F  4C  42  0A      00A8D          .ASCII  <10>\BLOCK_SPAN\
    54  4E  4F  43  5F  45  47  41  49  52  52  41  43      10      00A98          .ASCII  <16>\CARRIAGE_CONTROL\
                                            4C  4F  52      00      00AA6
    44  4C  45  49  46  5F  4C  4F  52  54  4E  4F  43      12      00AA9          .ASCII  <18>\CONTROL_FIELD_SIZE\
                                        45  5A  49  53  5F      00      00AB7
                            54  41  4D  52  4F  46  06      00ABC          .ASCII  <6>\FORMAT\
                                        45  5A  49  53  04      00AC3          .ASCII  <4>\SIZE\
                            45  54  45  4C  45  44  06      00AC8          .ASCII  <6>\DELETE\
                                            54  45  47  03      00ACF          .ASCII  <3>\GET\
                4D  41  45  52  54  53  49  54  4C  55  4D  0B      00AD3          .ASCII  <11>\MULTISTREAM\
                        54  49  42  49  48  4F  52  50  08      00ADF          .ASCII  <8>\PROHIBIT\
                                            54  55  50  03      00AE8          .ASCII  <3>\PUT\
                            45  54  41  44  50  55  06      00AEC          .ASCII  <6>\UPDATE\
    43  4F  4C  52  45  54  4E  49  5F  52  45  53  55  0E      00AF3          .ASCII  <14>\USER_INTERLOCK\
                                                        4B      00B01
                            45  43  49  56  45  44  06      00B02          .ASCII  <6>\DEVICE\
                            45  43  52  55  4F  53  06      00B09          .ASCII  <6>\SOURCE\
                        00  54  45  47  52  41  54  06      00B10          .ASCII  <6>\TARGET\<0>
            00000000  00000000  00000000  00000000  00000041      00B1C C.AAF:   .LONG   ^X41,0,0,0,0,0,0,0
                                00000000  00000000  00000000      00B30
                                        20  20  20  20      00B3C C.AAG:   .ASCII  \    \
                                        00  00  09  09      00B40 C.AAH:   .ASCII  <9><9><0><0>
                                        47  45  53  09      00B44 C.AAI:   .ASCII  <9>\SEG\
                        09  48  54  47  4E  45  4C  5F      00B48 C.AAJ:   .ASCII  \_LENGTH\<9>
                                        47  45  53  09      00B50 C.AAK:   .ASCII  <9>\SEG\
            00  00  09  4E  4F  49  54  49  53  4F  50  5F      00B54 C.AAL:   .ASCII  \_POSITION\<9><0><0>
```

```
                                00 09 09 45 50 59 54 09      00B60 C.AAM:   .ASCII  <9>\TYPE\<9><9><0>
00 72 65 64 6E 69 6C 79 63 5F 79 6E 61 09      00B68 C.AAN:   .ASCII  <9>\any_cylinder\<0><0><0>
                                00 00          00B76
                                00 72 65 74 73 75 6C 63      00B78 C.AAO:   .ASCII  \cluster\<0>
                                72 65 64 6E 69 6C 79 63      00B80 C.AAP:   .ASCII  \cylinder\
                                00 44 49 5F 65 6C 69 66      00B88 C.AAQ:   .ASCII  \file_ID\<0>
                     00 00 00 65 6D 61 6E 5F 65 6C 69 66      00B90 C.AAR:   .ASCII  \file_name\<0><0><0>
                                00 6C 61 63 69 67 6F 6C      00B9C C.AAS:   .ASCII  \logical\<0>
                                00 6C 61 75 74 72 69 76      00BA4 C.AAT:   .ASCII  \virtual\<0>
                                00 00 00 65 6E 6F 6E 09      00BAC C.AAU:   .ASCII  <9>\none\<0><0><0>
                                64 65 78 65 64 6E 69 09      00BB4 C.AAV:   .ASCII  <9>\indexed\
                     00 00 00 65 76 69 74 61 6C 65 72 09      00BBC C.AAW:   .ASCII  <9>\relative\<0><0><0>
                     00 6C 61 69 74 6E 65 75 71 65 73 09      00BC8 C.AAX:   .ASCII  <9>\sequential\<0>
72 65 76 6F 63 65 72 5F 6E 69 5F 66 69 09      00BD4 C.AAY:   .ASCII  <9>\if_in_recovery_unit\
                                74 69 6E 75 5F 79            00BE2
5F 6F 74 5F 79 72 61 73 73 65 63 65 6E 09      00BE8 C.AAZ:   .ASCII  <9>\necessary_to_write\<0>
                                00 65 74 69 72 77            00BF6
72 75 6F 6A 5F 55 52 5F 72 65 76 65 6E 09      00BFC C.ABA:   .ASCII  <9>\never_RU_journal\<0><0><0>
                                00 00 00 6C 61 6E            00C0A
75 74 65 72 5F 65 67 61 69 72 72 61 63 09      00C10 C.ABB:   .ASCII  <9>\carriage_return\
                                                6E 72        00C1E
                                4E 41 52 54 52 4F 46 09      00C20 C.ABC:   .ASCII  <9>\FORTRAN\
                                00 00 74 6E 69 72 70 09      00C28 C.ABD:   .ASCII  <9>\print\<0><0>
                                00 00 64 65 78 69 66 09      00C30 C.ABE:   .ASCII  <9>\fixed\<0><0>
                                00 6D 61 65 72 74 73 09      00C38 C.ABF:   .ASCII  <9>\stream\<0>
                     00 00 52 43 5F 6D 61 65 72 74 73 09      00C40 C.ABG:   .ASCII  <9>\stream_CR\<0><0>
                     00 00 46 4C 5F 6D 61 65 72 74 73 09      00C4C C.ABH:   .ASCII  <9>\stream_LF\<0><0>
                     00 00 64 65 6E 69 66 65 64 6E 75 09      00C58 C.ABI:   .ASCII  <9>\undefined\<0><0>
                     00 00 00 65 6C 62 61 69 72 61 76 09      00C64 C.ABJ:   .ASCII  <9>\variable\<0><0><0>
                                                43 46 56 09      00C70 C.ABK:   .ASCII  <9>\VFC\
                                00 00 00 32 6E 69 62 09      00C74 C.ABL:   .ASCII  <9>\bin2\<0><0><0>
                                00 00 00 34 6E 69 62 09      00C7C C.ABM:   .ASCII  <9>\bin4\<0><0><0>
                                00 00 00 38 6E 69 62 09      00C84 C.ABN:   .ASCII  <9>\bin8\<0><0><0>
                                6C 61 6D 69 63 65 64 09      00C8C C.ABO:   .ASCII  <9>\decimal\
                                00 00 00 32 74 6E 69 09      00C94 C.ABP:   .ASCII  <9>\int2\<0><0><0>
                                00 00 00 34 74 6E 69 09      00C9C C.ABQ:   .ASCII  <9>\int4\<0><0><0>
                                00 00 00 38 74 6E 69 09      00CA4 C.ABR:   .ASCII  <9>\int8\<0><0><0>
                                00 67 6E 69 72 74 73 09      00CAC C.ABS:   .ASCII  <9>\string\<0>
                                                53 41 49 09      00CB4 C.ABT:   .ASCII  <9>\IAS\
                                00 45 2F 53 54 53 52 09      00CB8 C.ABU:   .ASCII  <9>\RSTS/E\<0>
                                4D 31 31 2D 58 53 52 09      00CC0 C.ABV:   .ASCII  <9>\RSX-11M\
00 53 55 4C 50 2D 4D 31 31 2D 58 53 52 09      00CC8 C.ABW:   .ASCII  <9>\RSX-11M-PLUS\<0><0><0>
                                                00 00        00CD6
                                00 00 31 31 2D 54 52 09      00CD8 C.ABX:   .ASCII  <9>\RT-11\<0><0>
                                31 31 2D 58 41 52 54 09      00CE0 C.ABY:   .ASCII  <9>\TRAX-11\
                                53 4D 56 2F 58 41 56 09      00CE8 C.ABZ:   .ASCII  <9>\VAX/VMS\
                                                73 65 79 09      00CF0 C.ACA:   .ASCII  <9>\yes\
                                                00 6F 6E 09      00CF4 C.ACB:   .ASCII  <9>\no\<0>
                                                00 00 27 09      00CF8 C.ACC:   .ASCII  <9>\'\<0><0>
      00000000  00000000  00000000  00000000  000000A6      00CFC C.ACD:   .LONG   ^XA6,0,0,0,0,0,0,0
                          00000000  00000000  00000000      00D10
                                                00 00 28 09      00D1C C.ACE:   .ASCII  <9>\(\<0><0>
                                00 00 00 49 25 21 5F 21      00D20 C.ACF:   .ASCII  \! !%I\<0><0><0>
                     00 00 00 3A 6D 65 74 73 79 73 28 09      00D28 C.ACG:   .ASCII  <95\(system:\<0><0><0>
                                3A 72 65 6E 77 6F 20 2C      00D34 C.ACH:   .ASCII  \, owner:\
                                3A 70 75 6F 72 67 20 2C      00D3C C.ACI:   .ASCII  \, group:\
                                3A 64 6C 72 6F 77 20 2C      00D44 C.ACJ:   .ASCII  \, world:\
```

```
                                00000 CHECK_QUOTES:                                                     ; 0137
                            007C 00000          .WORD   ^M<R2,R3,R4,R5,R6>
                         5E      08   C2 00002          SUBL2   #8,SP
                         50      22   90 00005          MOVB    #34,QUOTES                             ; 0149
                              04 BC   B5 00008          TSTW    @4(R12)                                ; 0154
                                 03   12 0000B          BNEQ    .+3
                                0000V 31 0000D          BRW     29$
                         51      04   AC D0 00010          MOVL    4(R12),R1                          ; 0161
                         50      04   B1 91 00014          CMPB    @4(R1),QUOTES
                                 00V  12 00018          BNEQ    3$
                         51      04   BC 3C 0001A          MOVZWL  @4(R12),R1
                         52      04   AC D0 0001E          MOVL    4(R12),R2
                         52      04   A2 D0 00022          MOVL    4(R2),R2
                         50      FF A241 91 00026          CMPB    -1(R2)[R1],QUOTES
                                 03   12 0002B          BNEQ    .+3
                                0000V 31 0002D          BRW     30$
                         52      04   AC D0 00030 3$:      MOVL    4(R12),R2
                00000000G EF     04   B2 91 00034          CMPB    @4(R2),APOSTROPHE
                                 00V  12 0003C          BNEQ    5$
                         52      04   BC 3C 0003E          MOVZWL  @4(R12),R2
                         51      04   AC D0 00042          MOVL    4(R12),R1
                         51      04   A1 D0 00046          MOVL    4(R1),R1
                00000000G EF     FF A142 91 0004A          CMPB    -1(R1)[R2],APOSTROPHE
                                 03   12 00053          BNEQ    .+3
                                0000V 31 00055          BRW     30$
                                 51   94 00058 5$:      CLRB    QUOTE_FOUND                            ; 0180
                                 52   94 0005A          CLRB    APOST_FOUND                            ; 0181
                         53      01   D0 0005C          MOVL    #1,R3                                  ; 0183
                         54      04   BC 3C 0005F          MOVZWL  @4(R12),R4
                         54           53   D1 00063          CMPL    R3,R4
                                 00V  14 00066          BGTR    11$
                         55           53   D0 00068 6$:      MOVL    R3,INDEX
                         56      04   AC D0 0006B          MOVL    4(R12),R6                          ; 0187
                         56      04   A6 D0 0006F          MOVL    4(R6),R6
                         50      FF A645 91 00073          CMPB    -1(R6)[INDEX],QUOTES
                                 00V  12 00078          BNEQ    8$
                         51           01   90 0007A          MOVB    #1,QUOTE_FOUND                     ; 0189
                         56      04   AC D0 0007D 8$:      MOVL    4(R12),R6                          ; 0191
                         56      04   A6 D0 00081          MOVL    4(R6),R6
                00000000G EF     FF A645 91 00085          CMPB    -1(R6)[INDEX],APOSTROPHE
                                 00V  12 0008E          BNEQ    10$
                         52           01   90 00090          MOVB    #1,APOST_FOUND                     ; 0193
                D1       53           54   F3 00093 10$:      AOBLEQ  R4,R3,6$
                         00V          51   E9 00097 11$:      BLBC    QUOTE_FOUND,24$                    ; 0202
                         00V          52   E9 0009A          BLBC    APOST_FOUND,22$                    ; 0206
                         52      04   BC 3C 0009D          MOVZWL  @4(R12),SCAN_INDEX                 ; 0210
                         51      04   AC D0 000A1 14$:      MOVL    4(R12),R1                          ; 0214
                         51      04   A1 D0 000A5          MOVL    4(R1),R1
                         50      FF A142 91 000A9          CMPB    -1(R1)[SCAN_INDEX],QUOTES
                                 00V  12 000AE          BNEQ    20$
                         51      04   BC 3C 000B0          MOVZWL  @4(R12),R1                         ; 0218
                         53           52   D0 000B4          MOVL    SCAN_INDEX,R3
                         53           51   D1 000B7          CMPL    R1,R3
                                 00V  19 000BA          BLSS    17$
                         55           51   D0 000BC 16$:      MOVL    R1,INDEX
                         54      04   AC D0 000BF          MOVL    4(R12),R4                          ; 0220
```

```
                                56        04   AC   D0 000C3          MOVL    4(R12),R6
                                56        04   A6   D0 000C7          MOVL    4(R6),R6
                          04 B445         FF A645   90 000CB          MOVB    -1(R6)[INDEX],a4(R4)[INDEX]
          FFE0  51 FFFFFFFF   8F              53   F1 000D2          ACBL    R3,#-1,R1,16$
                      00FE    8F        04    BC   B1 000DC 17$:     CMPW    a4(R12),#254                                    ; 0223
                                              00V  1E 000E2          BGEQU   20$
                                        04    BC   B6 000E4          INCW    a4(R12)                                         ; 0225
                                              52   D7 000E7 20$:     DECL    SCAN_INDEX                                      ; 0229
                                01            52   D1 000E9          CMPL    SCAN_INDEX,#1
                                              B3   18 000EC          BGEQ    14$
                                52            50   90 000EE          MOVB    QUOTES,QCHAR                                    ; 0233
                                              00V  11 000F1          BRB     25$
                                52 00000000G  EF   90 000F3 22$:     MOVB    APOSTROPHE,QCHAR                                ; 0241
                                              00V  11 000FA          BRB     25$
                                52            50   90 000FC 24$:     MOVB    QUOTES,QCHAR                                    ; 0251
                                50        04    BC   3C 000FF 25$:     MOVZWL  a4(R12),R0                                      ; 0258
                                              00V  15 00103          BLEQ    27$
                                55            50   D0 00105 26$:     MOVL    R0,INDEX
                                51        04    AC   D0 00108          MOVL    4(R12),R1                                       ; 0260
                                53        04    AC   D0 0010C          MOVL    4(R12),R3
                                53        04    A3   D0 00110          MOVL    4(R3),R3
                          04 B145         FF A345   90 00114          MOVB    -1(R3)[INDEX],a4(R1)[INDEX]
                                E7              50   F5 0011B          SOBGTR  R0,26$
                          04    BC              02   A0 0011E 27$:     ADDW2   #2,a4(R12)                                      ; 0265
                                50        04    AC   D0 00122          MOVL    4(R12),R0                                       ; 0266
                          04    B0              52   90 00126          MOVB    QCHAR,a4(R0)
                                50        04    BC   3C 0012A          MOVZWL  a4(R12),R0                                      ; 0267
                                55        04    AC   D0 0012E          MOVL    4(R12),R5
                                55        04    A5   D0 00132          MOVL    4(R5),R5
                          FF A540               52   90 00136          MOVB    QCHAR,-1(R5)[R0]
                                              00V  11 0013B          BRB     30$
                          F8    AD 010E0002     8F   D0 0013D 29$:     MOVL    #17694722,-8(FP)                                ; 0277
                          FC    AD 00000000G    EF   9E 00145          MOVAB   EMPTY_STRING,-4(FP)
                                      F8        AD   9F 0014D          PUSHAB  -8(FP)
                                      04        AC   DD 00150          PUSHL   4(R12)
                    00000000G   EF              02   FB 00153          CALLS   #2,STR$TRIM
                                                04 0015A 30$:     RET                                                     ; 0281

; Routine Size: 347 bytes,    Routine Base: $CODE + 00D4C

                                                   00000 SHOW_PRIMARY:                                                   ; 0328
                                              0004 00000          .WORD   ^M<R2>
                                5C 00000000G   EF   D0 00002          MOVL    DEF_CURRENT,R12                                 ; 0335
                                52       19    AC   90 00009          MOVB    25(R12),TEMP_PRI                                ; 0342
                                50            52   9A 0000D          MOVZBL  TEMP_PRI,R0                                     ; 0347
                                7E 00000000GEF40   9A 00010          MOVZBL  PRIMARY_WIDTH[R0],-(SP)
                                7E            52   9A 00018          MOVZBL  TEMP_PRI,-(SP)
                                   FFFFF138   EF   9F 0001B          PUSHAB  C.AAA
                                   00000000G  EF   9F 00021          PUSHAB  FDL_DEST
                    00000000G   EF              04   FB 00027          CALLS   #4,PASSWRITE_ENUMERATED
                                50            52   9A 0002E          MOVZBL  TEMP_PRI,R0                                     ; 0352
                                10            50   D1 00031          CMPL    R0,#16
                                              00V  1E 00034          BGEQU   3$
                    00VFFFFF1F3   EF            50   E1 00036          BBC     R0,C.AAB,3$
                                              01   DD 0003E          PUSHL   #1                                              ; 0354
                                              20   DD 00040          PUSHL   #32
                                   00000000G  EF   9F 00042          PUSHAB  FDL_DEST
```

```
                    00000000G  EF                 03  FB 00048          CALLS   #3,PAS$WRITE_CHAR
                                             1A    AC  9F 0004F          PUSHAB  26(R12)
                    00000000G  EF                 01  FB 00052          CALLS   #1,NUM_LEN
                                                  50  DD 00059          PUSHL   R0
                                             1A    AC  DD 0005B          PUSHL   26(R12)
                                       00000000G   EF  9F 0005E          PUSHAB  FDL_DEST
                    00000000G  EF                 03  FB 00064          CALLS   #3,PAS$WRITE_INTEGER
                               52                 52  9A 0006B 3$:      MOVZBL  TEMP_PRI,R2                                 ; 0359
                               10                 52  D1 0006E          CMPL    R2,#T6
                                                  00V 1E 00071          BGEQU   7$
                00VFFFFF1B8    EF                 52  E1 00073          BBC     R2,C.AAC,7$
                                             11    AC  9F 0007B          PUSHAB  17(R12)                                     ; 0363
                      0D4C     CF                 01  FB 0007E          CALLS   #1,CHECK_QUOTES
                                             11    AC  B5 00083          TSTW    17(R12)                                     ; 0365
                                                  00V 15 00086          BLEQ    7$
                                                  01  DD 00088          PUSHL   #1                                          ; 0367
                                                  09  DD 0008A          PUSHL   #9
                                       00000000G   EF  9F 0008C          PUSHAB  FDL_DEST
                    00000000G  EF                 03  FB 00092          CALLS   #3,PAS$WRITE_CHAR
                               7E             11    AC  3C 00099          MOVZWL  17(R12),-(SP)
                                                  00  DD 0009D          PUSHL   #0
                                             15    BC  9F 0009F          PUSHAB  @21(R12)
                                          000000FF   8F  DD 000A2          PUSHL   #255
                                       00000000G   EF  9F 000A8          PUSHAB  FDL_DEST
                    00000000G  EF                 05  FB 000AE          CALLS   #5,PAS$WRITE_STRING
                                                  04  00B5 7$:          RET                                                 ; 0374

; Routine Size: 182 bytes,    Routine Base: $CODE + 00EA7

                                                 00000 SHOW_SECONDARY:                                                    ; 0421
                                                  000C 00000            .WORD   ^M<R2,R3>
                               5C 00000000G   EF  D0  00002            MOVL    DEF_CURRENT,R12                             ; 0428
                               52       1E    AC  90  00009            MOVB    30(R12),TEMP_SEC                            ; 0435
                               50             52  9A  0000D            MOVZBL  TEMP_SEC,R0                                 ; 0440
                         00000098     8F       50  D1  00010            CMPL    R0,#T52
                                               00V 1E  00017            BGEQU   2$
                    03 FFFFF15E     EF         50  E1  00019            BBC     R0,C.AAD,.+3
                                             0000V 31  00021            BRW     12$
                                               01  DD  00024 2$:        PUSHL   #1                                          ; 0444
                                               09  DD  00026            PUSHL   #9
                                       00000000G   EF  9F  00028        PUSHAB  FDL_DEST
                        00000000G    EF           03  FB  0002E        CALLS   #3,PAS$WRITE_CHAR
                                     50           52  9A  00035        MOVZBL  TEMP_SEC,R0
                                     53 00000000GEF40  9E  00038        MOVAB   SECONDARY_WIDTH[R0],R3
                                     7E           63  9A  00040        MOVZBL  (R3),-(SP)
                                     7E           52  9A  00043        MOVZBL  TEMP_SEC,-(SP)
                                          FFFFF147   EF  9F  00046        PUSHAB  C.AAE
                                       00000000G   EF  9F  0004C        PUSHAB  FDL_DEST
                        00000000G    EF           04  FB  00052        CALLS   #4,PAS$WRITE_ENUMERATED
                                     21           52  91  00059        CMPB    TEMP_SEC,#33                                ; 0449
                                                 00V 12  0005C        BNEQ    5$
                         00000100    8F       23  AC  D1  0005E        CMPL    35(R12),#256
                                                 00V 1E  00066        BGEQU   4$
                    00VFFFFFB4E     EF       23  AC  E0  00068        BBS     35(R12),C.AAF,5$
                                          FFFFFB68   EF  9F  00071 4$:    PUSHAB  C,AAG                                       ; 0455
                                                  04  DD  00077        PUSHL   #4
                                       00000000G   EF  9F  00079        PUSHAB  FDL_DEST
```

```
00000000G  EF  03  FB 0007F        CALLS   #3,PAS$WRITE_STRING
         0000V 31  00086           BRW     19$
      08   63  91  00089 5$:       CMPB    (R3),#8                          ; 0457
           00V 1E  0008C           BGEQU   7$
   FFFFFB4F   EF  9F  0008E        PUSHAB  C.AAH                            ; 0459
           02  DD  00094           PUSHL   #2
   00000000G  EF  9F  00096        PUSHAB  FDL_DEST
00000000G  EF  03  FB 0009C        CALLS   #3,PAS$WRITE_STRING
         0000V 31  000A3           BRW     19$
      10   63  91  000A6 7$:       CMPB    (R3),#16                         ; 0461
           03  1F  000A9           BLSSU   .+3
         0000V 31  000AB           BRW     19$
           01  DD  000AE           PUSHL   #1                               ; 0463
           09  DD  000B0           PUSHL   #9
   00000000G  EF  9F  000B2        PUSHAB  FDL_DEST
00000000G  EF  03  FB 000B8        CALLS   #3,PAS$WRITE_CHAR
         0000V 31  000BF           BRW     19$
      85   8F  52  91  000C2 12$:  CMPB    TEMP_SEC,#-123                   ; 0474
           00V 12  000C6           BNEQ    14$
   FFFFFB19   EF  9F  000C8        PUSHAB  C.AAI                            ; 0476
           04  DD  000CE           PUSHL   #4
   00000000G  EF  9F  000D0        PUSHAB  FDL_DEST
00000000G  EF  03  FB 000D6        CALLS   #3,PAS$WRITE_STRING
           01  DD  000DD           PUSHL   #1
         1F   AC  DD  000DF        PUSHL   31(R12)
   00000000G  EF  9F  000E2        PUSHAB  FDL_DEST
00000000G  EF  03  FB 000E8        CALLS   #3,PAS$WRITE_INTEGER
   FFFFFAF6   EF  9F  000EF        PUSHAB  C.AAJ
           08  DD  000F5           PUSHL   #8
   00000000G  EF  9F  000F7        PUSHAB  FDL_DEST
00000000G  EF  03  FB 000FD        CALLS   #3,PAS$WRITE_STRING
      86   8F  52  91  00104 14$:  CMPB    TEMP_SEC,#-122                   ; 0481
           00V 12  00108           BNEQ    16$
   FFFFFAE3   EF  9F  0010A        PUSHAB  C.AAK                            ; 0483
           04  DD  00110           PUSHL   #4
   00000000G  EF  9F  00112        PUSHAB  FDL_DEST
00000000G  EF  03  FB 00118        CALLS   #3,PAS$WRITE_STRING
           01  DD  0011F           PUSHL   #1
         1F   AC  DD  00121        PUSHL   31(R12)
   00000000G  EF  9F  00124        PUSHAB  FDL_DEST
00000000G  EF  03  FB 0012A        CALLS   #3,PAS$WRITE_INTEGER
   FFFFFAC0   EF  9F  00131        PUSHAB  C.AAL
           0A  DD  00137           PUSHL   #10
   00000000G  EF  9F  00139        PUSHAB  FDL_DEST
00000000G  EF  03  FB 0013F        CALLS   #3,PAS$WRITE_STRING
      87   8F  52  91  00146 16$:  CMPB    TEMP_SEC,#-121                   ; 0488
           00V 12  0014A           BNEQ    19$
   FFFFFAB1   EF  9F  0014C        PUSHAB  C.AAM                            ; 0490
           07  DD  00152           PUSHL   #7
   00000000G  EF  9F  00154        PUSHAB  FDL_DEST
00000000G  EF  03  FB 0015A        CALLS   #3,PAS$WRITE_STRING
                   04  00161 19$:  RET                                      ; 0502

; Routine Size: 354 bytes,    Routine Base: $CODE + 00F5D

                         00000 SHOW_QUALIFIER:                              ; 0548
                    0000 00000          .WORD   ^M<>
```

```
                    50 00000000G  EF    DO 00002          MOVL    DEF_CURRENT,RO                                  ; 0555
                28  00       23   AO    CF 00009          CASEL   35(RO),#0,#40
                                    0000V     0000E         .DISPL  1$
                                    0000V     00010         .DISPL  2$
                                    0000V     00012         .DISPL  3$
                                    0000V     00014         .DISPL  4$
                                    0000V     00016         .DISPL  5$
                                    0000V     00018         .DISPL  6$
                                    0000V     0001A         .DISPL  8$
                                    0000V     0001C         .DISPL  7$
                                    0000V     0001E         .DISPL  8$
                                    0000V     00020         .DISPL  15$
                                    0000V     00022         .DISPL  16$
                                    0000V     00024         .DISPL  17$
                                    0000V     00026         .DISPL  22$
                                    0000V     00028         .DISPL  18$
                                    0000V     0002A         .DISPL  23$
                                    0000V     0002C         .DISPL  24$
                                    0000V     0002E         .DISPL  19$
                                    0000V     00030         .DISPL  21$
                                    0000V     00032         .DISPL  20$
                                    0000V     00034         .DISPL  12$
                                    0000V     00036         .DISPL  13$
                                    0000V     00038         .DISPL  14$
                                    0000V     0003A         .DISPL  33$
                                    0000V     0003C         .DISPL  34$
                                    0000V     0003E         .DISPL  35$
                                    0000V     00040         .DISPL  36$
                                    0000V     00042         .DISPL  37$
                                    0000V     00044         .DISPL  38$
                                    0000V     00046         .DISPL  39$
                                    0000V     00048         .DISPL  11$
                                    0000V     0004A         .DISPL  10$
                                    0000V     0004C         .DISPL  9$
                                    0052      0004E         .DISPL  82
                                    0000V     00050         .DISPL  32$
                                    0000V     00052         .DISPL  29$
                                    0000V     00054         .DISPL  25$
                                    0000V     00056         .DISPL  30$
                                    0000V     00058         .DISPL  26$
                                    0000V     0005A         .DISPL  31$
                                    0000V     0005C         .DISPL  27$
                                    0000V     0005E         .DISPL  28$
                                    0000V  31 00060         BRW     40$
                     FFFFFA40    EF    9F 00063 1$:        PUSHAB  C.AAN                                           ; 0557
                                 0D    DD 00069            PUSHL   #13
                     00000000G   EF    9F 0006B            PUSHAB  FDL_DEST
00000000G   EF                   03    FB 00071            CALLS   #3,PAS$WRITE_STRING
                                    0000V  31 00078         BRW     41$
                     FFFFFA38    EF    9F 0007B 2$:        PUSHAB  C.AAO                                           ; 0558
                                 07    DD 00081            PUSHL   #7
                     00000000G   EF    9F 00083            PUSHAB  FDL_DEST
00000000G   EF                   03    FB 00089            CALLS   #3,PAS$WRITE_STRING
                                    0000V  31 00090         BRW     41$
                     FFFFFA28    EF    9F 00093 3$:        PUSHAB  C.AAP                                           ; 0559
                                 08    DD 00099            PUSHL   #8
                     00000000G   EF    9F 0009B            PUSHAB  FDL_DEST
```

```
00000000G  EF  03  FB 000A1        CALLS  #3,PAS$WRITE_STRING
          0000V 31  000A8          BRW    41$
    FFFFFA18  EF  9F  000AB 4$:    PUSHAB C.AAQ                          ; 0560
              07  DD  000B1       PUSHL  #7
    00000000G EF  9F  000B3        PUSHAB FDL_DEST
00000000G  EF  03  FB 000B9        CALLS  #3,PAS$WRITE_STRING
          0000V 31  000C0          BRW    41$
    FFFFFA08  EF  9F  000C3 5$:    PUSHAB C.AAR                          ; 0561
              09  DD  000C9       PUSHL  #9
    00000000G EF  9F  000CB        PUSHAB FDL_DEST
00000000G  EF  03  FB 000D1        CALLS  #3,PAS$WRITE_STRING
          0000V 31  000D8          BRW    41$
    FFFFF9FC  EF  9F  000DB 6$:    PUSHAB C.AAS                          ; 0562
              07  DD  000E1       PUSHL  #7
    00000000G EF  9F  000E3        PUSHAB FDL_DEST
00000000G  EF  03  FB 000E9        CALLS  #3,PAS$WRITE_STRING
          0000V 31  000F0          BRW    41$
    FFFFF9EC  EF  9F  000F3 7$:    PUSHAB C.AAT                          ; 0563
              07  DD  000F9       PUSHL  #7
    00000000G EF  9F  000FB        PUSHAB FDL_DEST
00000000G  EF  03  FB 00101        CALLS  #3,PAS$WRITE_STRING
          0000V 31  00108          BRW    41$
    FFFFF9DC  EF  9F  0010B 8$:    PUSHAB C.AAU                          ; 0565
              05  DD  00111       PUSHL  #5
    00000000G EF  9F  00113        PUSHAB FDL_DEST
00000000G  EF  03  FB 00119        CALLS  #3,PAS$WRITE_STRING
          0000V 31  00120          BRW    41$
    FFFFF9CC  EF  9F  00123 9$:    PUSHAB C.AAV                          ; 0566
              08  DD  00129       PUSHL  #8
    00000000G EF  9F  0012B        PUSHAB FDL_DEST
00000000G  EF  03  FB 00131        CALLS  #3,PAS$WRITE_STRING
          0000V 31  00138          BRW    41$
    FFFFF9BC  EF  9F  0013B 10$:   PUSHAB C.AAW                          ; 0567
              09  DD  00141       PUSHL  #9
    00000000G EF  9F  00143        PUSHAB FDL_DEST
00000000G  EF  03  FB 00149        CALLS  #3,PAS$WRITE_STRING
          0000V 31  00150          BRW    41$
    FFFFF9B0  EF  9F  00153 11$:   PUSHAB C.AAX                          ; 0568
              0B  DD  00159       PUSHL  #11
    00000000G EF  9F  0015B        PUSHAB FDL_DEST
00000000G  EF  03  FB 00161        CALLS  #3,PAS$WRITE_STRING
          0000V 31  00168          BRW    41$
    FFFFF9A4  EF  9F  0016B 12$:   PUSHAB C.AAY                          ; 0569
              14  DD  00171       PUSHL  #20
    00000000G EF  9F  00173        PUSHAB FDL_DEST
00000000G  EF  03  FB 00179        CALLS  #3,PAS$WRITE_STRING
          0000V 31  00180          BRW    41$
    FFFFF9A0  EF  9F  00183 13$:   PUSHAB C.AAZ                          ; 0570
              13  DD  00189       PUSHL  #19
    00000000G EF  9F  0018B        PUSHAB FDL_DEST
00000000G  EF  03  FB 00191        CALLS  #3,PAS$WRITE_STRING
          0000V 31  00198          BRW    41$
    FFFFF99C  EF  9F  0019B 14$:   PUSHAB C.ABA                          ; 0571
              11  DD  001A1       PUSHL  #17
    00000000G EF  9F  001A3        PUSHAB FDL_DEST
00000000G  EF  03  FB 001A9        CALLS  #3,PAS$WRITE_STRING
          0000V 31  001B0          BRW    41$
```

```
                FFFFF998   EF   9F 001B3 15$:    PUSHAB  C.ABB                                  ; 0572
                           10   DD 001B9         PUSHL   #16
                00000000G  EF   9F 001BB         PUSHAB  FDL_DEST
00000000G  EF              03   FB 001C1         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 001C8         BRW     41$
                FFFFF990   EF   9F 001CB 16$:    PUSHAB  C.ABC                                  ; 0573
                           08   DD 001D1         PUSHL   #8
                00000000G  EF   9F 001D3         PUSHAB  FDL_DEST
00000000G  EF              03   FB 001D9         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 001E0         BRW     41$
                FFFFF980   EF   9F 001E3 17$:    PUSHAB  C.ABD                                  ; 0574
                           06   DD 001E9         PUSHL   #6
                00000000G  EF   9F 001EB         PUSHAB  FDL_DEST
00000000G  EF              03   FB 001F1         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 001F8         BRW     41$
                FFFFF970   EF   9F 001FB 18$:    PUSHAB  C.ABE                                  ; 0575
                           06   DD 00201         PUSHL   #6
                00000000G  EF   9F 00203         PUSHAB  FDL_DEST
00000000G  EF              03   FB 00209         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 00210         BRW     41$
                FFFFF960   EF   9F 00213 19$:    PUSHAB  C.ABF                                  ; 0576
                           07   DD 00219         PUSHL   #7
                00000000G  EF   9F 0021B         PUSHAB  FDL_DEST
00000000G  EF              03   FB 00221         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 00228         BRW     41$
                FFFFF950   EF   9F 0022B 20$:    PUSHAB  C.ABG                                  ; 0577
                           0A   DD 00231         PUSHL   #10
                00000000G  EF   9F 00233         PUSHAB  FDL_DEST
00000000G  EF              03   FB 00239         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 00240         BRW     41$
                FFFFF944   EF   9F 00243 21$:    PUSHAB  C.ABH                                  ; 0578
                           0A   DD 00249         PUSHL   #10
                00000000G  EF   9F 0024B         PUSHAB  FDL_DEST
00000000G  EF              03   FB 00251         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 00258         BRW     41$
                FFFFF938   EF   9F 0025B 22$:    PUSHAB  C.ABI                                  ; 0579
                           0A   DD 00261         PUSHL   #10
                00000000G  EF   9F 00263         PUSHAB  FDL_DEST
00000000G  EF              03   FB 00269         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 00270         BRW     41$
                FFFFF92C   EF   9F 00273 23$:    PUSHAB  C.ABJ                                  ; 0580
                           09   DD 00279         PUSHL   #9
                00000000G  EF   9F 0027B         PUSHAB  FDL_DEST
00000000G  EF              03   FB 00281         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 00288         BRW     41$
                FFFFF920   EF   9F 0028B 24$:    PUSHAB  C.ABK                                  ; 0581
                           04   DD 00291         PUSHL   #4
                00000000G  EF   9F 00293         PUSHAB  FDL_DEST
00000000G  EF              03   FB 00299         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 002A0         BRW     41$
                FFFFF90C   EF   9F 002A3 25$:    PUSHAB  C.ABL                                  ; 0582
                           05   DD 002A9         PUSHL   #5
                00000000G  EF   9F 002AB         PUSHAB  FDL_DEST
00000000G  EF              03   FB 002B1         CALLS   #3,PAS$WRITE_STRING
                         0000V  31 002B8         BRW     41$
                FFFFF8FC   EF   9F 002BB 26$:    PUSHAB  C.ABM                                  ; 0583
                           05   DD 002C1         PUSHL   #5
```

```
                    00000000G   EF   9F 002C3          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 002C9          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   002D0          BRW     41$
                    FFFFF8EC    EF   9F 002D3 27$:     PUSHAB  C.ABN                                   ; 0584
                                05   DD 002D9          PUSHL   #5
                    00000000G   EF   9F 002DB          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 002E1          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   002E8          BRW     41$
                    FFFFF8DC    EF   9F 002EB 28$:     PUSHAB  C.ABO                                   ; 0585
                                08   DD 002F1          PUSHL   #8
                    00000000G   EF   9F 002F3          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 002F9          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   00300          BRW     41$
                    FFFFF8CC    EF   9F 00303 29$:     PUSHAB  C.ABP                                   ; 0586
                                05   DD 00309          PUSHL   #5
                    00000000G   EF   9F 0030B          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 00311          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   00318          BRW     41$
                    FFFFF8BC    EF   9F 0031B 30$:     PUSHAB  C.ABQ                                   ; 0587
                                05   DD 00321          PUSHL   #5
                    00000000G   EF   9F 00323          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 00329          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   00330          BRW     41$
                    FFFFF8AC    EF   9F 00333 31$:     PUSHAB  C.ABR                                   ; 0588
                                05   DD 00339          PUSHL   #5
                    00000000G   EF   9F 0033B          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 00341          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   00348          BRW     41$
                    FFFFF89C    EF   9F 0034B 32$:     PUSHAB  C.ABS                                   ; 0589
                                07   DD 00351          PUSHL   #7
                    00000000G   EF   9F 00353          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 00359          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   00360          BRW     41$
                    FFFFF88C    EF   9F 00363 33$:     PUSHAB  C.ABT                                   ; 0590
                                04   DD 00369          PUSHL   #4
                    00000000G   EF   9F 0036B          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 00371          CALLS   #3,PAS$WRITE_STRING
                          0000V 31   00378          BRW     41$
                    FFFFF878    EF   9F 0037B 34$:     PUSHAB  C.ABU                                   ; 0591
                                07   DD 00381          PUSHL   #7
                    00000000G   EF   9F 00383          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 00389          CALLS   #3,PAS$WRITE_STRING
                          00V   11   00390          BRB     41$
                    FFFFF869    EF   9F 00392 35$:     PUSHAB  C.ABV                                   ; 0592
                                08   DD 00398          PUSHL   #8
                    00000000G   EF   9F 0039A          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 003A0          CALLS   #3,PAS$WRITE_STRING
                          00V   11   003A7          BRB     41$
                    FFFFF85A    EF   9F 003A9 36$:     PUSHAB  C.ABW                                   ; 0593
                                0D   DD 003AF          PUSHL   #13
                    00000000G   EF   9F 003B1          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 003B7          CALLS   #3,PAS$WRITE_STRING
                          00V   11   003BE          BRB     41$
                    FFFFF853    EF   9F 003C0 37$:     PUSHAB  C.ABX                                   ; 0594
                                06   DD 003C6          PUSHL   #6
                    00000000G   EF   9F 003C8          PUSHAB  FDL_DEST
00000000G   EF                  03   FB 003CE          CALLS   #3,PAS$WRITE_STRING
```

```
                                    00V  11 003D5          BRB     41$
                          FFFFF844  EF   9F 003D7 38$:     PUSHAB  C.ABY                                   ; 0595
                                    08   DD 003DD          PUSHL   #8
                          00000000G EF   9F 003DF          PUSHAB  FDL_DEST
                00000000G EF              03   FB 003E5    CALLS   #3,PAS$WRITE_STRING
                                    00V  11 003EC          BRB     41$
                          FFFFF835  EF   9F 003EE 39$:     PUSHAB  C.ABZ                                   ; 0596
                                    08   DD 003F4          PUSHL   #8
                          00000000G EF   9F 003F6          PUSHAB  FDL_DEST
                00000000G EF              03   FB 003FC    CALLS   #3,PAS$WRITE_STRING
                                    00V  11 00403          BRB     41$
                                         00405 40$:
                                    04   00405 41$:        RET                                             ; 0604

; Routine Size: 1030 bytes,    Routine Base: $CODE + 010BF

                                         00000 SHOW_CURRENT:                                             ; 0656
                                    0004 00000             .WORD   ^M<R2>
                          5E             14   C2 00002    SUBL2   #20,SP
                          5C       04    BC   90 00005    MOVB    @4(R12),PLUS_VALUE
              00V00000000G EF              00   E1 00009    BBC     #0,DEST_IS_TERMINAL,2$                  ; 0688
                          00000000G EF   9F 00011          PUSHAB  SHIFT                                   ; 0690
                                    04   DD 00017          PUSHL   #4
                          00000000G EF   9F 00019          PUSHAB  FDL_DEST
                00000000G EF              03   FB 0001F    CALLS   #3,PAS$WRITE_STRING
                          50 00000000G EF   D0 00026 2$:     MOVL    DEF_CURRENT,R0                       ; 0695
                          50             60   9A 0002D    MOVZBL  (R0),R0
                02        00             50   8F 00030    CASEB   R0,#0,#2
                                    0000V     00034          .DISPL  3$
                                    0000V     00036          .DISPL  9$
                                    0000V     00038          .DISPL  51$
                                    0000V  31 0003A          BRW     55$
                          52 00000000G EF   D0 0003D 3$:     MOVL    DEF_CURRENT,R2                       ; 0699
                    0EA7  CF                 00   FB 00044    CALLS   #0,SHOW_PRIMARY                      ; 0706
                          00V                5C   E9 00049    BLBC    PLUS_VALUE,8$                        ; 0711
                                       09    A2   B5 0004C    TSTW    9(R2)                                ; 0716
                                    00V  15 0004F          BLEQ    8$
                                    01   DD 00051          PUSHL   #1                                   ; 0718
                                    09   DD 00053          PUSHL   #9
                          00000000G EF   9F 00055          PUSHAB  FDL_DEST
                00000000G EF              03   FB 0005B    CALLS   #3,PAS$WRITE_CHAR
                          7E       09    A2   3C 00062    MOVZWL  9(R2),-(SP)
                                    00   DD 00066          PUSHL   #0
                                    0D   B2   9F 00068     PUSHAB  @13(R2)
                          000000FF  8F   DD 0006B          PUSHL   #255
                          00000000G EF   9F 00071          PUSHAB  FDL_DEST
                00000000G EF              05   FB 00077    CALLS   #5,PAS$WRITE_STRING
                          00000000G EF   9F 0007E 8$:      PUSHAB  FDL_DEST                                ; 0724
                00000000G EF              01   FB 00084    CALLS   #1,PAS$WRITELN2
                                    0000V  31 0008B          BRW     56$
                          52 00000000G EF   D0 0008E 9$:     MOVL    DEF_CURRENT,R2                       ; 0730
                    0F5D  CF                 00   FB 00095    CALLS   #0,SHOW_SECONDARY                    ; 0737
                          03                 5C   E8 0009A    BLBS    PLUS_VALUE,.+3                       ; 0742
                                    0000V  31 0009D          BRW     50$
                          5C       1E    A2   9A 000A0     MOVZBL  30(R2),R12                           ; 0749
                          5C             04   C4 000A4     MULL2   #4,R12
              00V00000000G EF              5C   E1 000A7    BBC     R12,SEC_TYPE,15$
```

```
                                11   A2   9F 000AF          PUSHAB  17(R2)                                  ; 0753
                            0D4C CF      01 FB 000B2          CALLS   #1,CHECK_QUOTES
                                11   A2   B5 000B7          TSTW    17(R2)                                  ; 0755
                                     00V  15 000BA          BLEQ    15$
                                     01   DD 000BC          PUSHL   #1                                      ; 0757
                                     09   DD 000BE          PUSHL   #9
                           00000000G EF   9F 000C0          PUSHAB  FDL_DEST
                        00000000G    EF   03 FB 000C6          CALLS   #3,PAS$WRITE_CHAR
                                7E   11   A2 3C 000CD          MOVZWL  17(R2),-(SP)
                                     00   DD 000D1          PUSHL   #0
                                15   B2   9F 000D3          PUSHAB  @21(R2)
                           000000FF  8F   DD 000D6          PUSHL   #255
                           00000000G EF   9F 000DC          PUSHAB  FDL_DEST
                        00000000G    EF   05 FB 000E2          CALLS   #5,PAS$WRITE_STRING
                                50   1E   A2 9A 000E9 15$:    MOVZBL  30(R2),R0                               ; 0765
                                50        04 C4 000ED          MULL2   #4,R0
                                50        02 C0 000F0          ADDL2   #2,R0
                     00V00000000G EF        50 E1 000F3          BBC     R0,SEC_TYPE,17$
                           10BF   CF        00 FB 000FB          CALLS   #0,SHOW_QUALIFIER                       ; 0767
                                50   1E   A2 9A 00100 17$:    MOVZBL  30(R2),R0                               ; 0772
                                50        04 C4 00104          MULL2   #4,R0
                                          50 D6 00107          INCL    R0
                     00V00000000G    EF   50 E1 00109          BBC     R0,SEC_TYPE,19$
                                          01 DD 00111          PUSHL   #1                                      ; 0777
                                          09 DD 00113          PUSHL   #9
                                00000000G EF 9F 00115          PUSHAB  FDL_DEST
                             00000000G    EF 03 FB 0011B          CALLS   #3,PAS$WRITE_CHAR
                                     27   A2 9F 00122          PUSHAB  39(R2)
                             00000000G    EF 01 FB 00125          CALLS   #1,NUM_LEN
                                          50 DD 0012C          PUSHL   R0
                                     27   A2 DD 0012E          PUSHL   39(R2)
                                00000000G EF 9F 00131          PUSHAB  FDL_DEST
                             00000000G    EF 03 FB 00137          CALLS   #3,PAS$WRITE_INTEGER
                                     50   1E A2 9A 0013E 19$:    MOVZBL  30(R2),R0                           ; 0782
                                     50      04 C4 00142          MULL2   #4,R0
                                     50      03 C0 00145          ADDL2   #3,R0
                          00V00000000G EF      50 E1 00148          BBC     R0,SEC_TYPE,24$
                          00V       2B A2      00 E1 00150          BBC     #0,43(R2),22$                       ; 0784
                                       FFFFF6D0 EF 9F 00155          PUSHAB  C.ACA                               ; 0786
                                          04    DD 0015B          PUSHL   #4
                                00000000G EF    9F 0015D          PUSHAB  FDL_DEST
                             00000000G    EF    03 FB 00163          CALLS   #3,PAS$WRITE_STRING
                                          00V   11 0016A          BRB     24$
                                       FFFFF6BD EF 9F 0016C 22$:    PUSHAB  C.ACB                               ; 0790
                                          03    DD 00172          PUSHL   #3
                                00000000G EF    9F 00174          PUSHAB  FDL_DEST
                             00000000G    EF    03 FB 0017A          CALLS   #3,PAS$WRITE_STRING
                                     83   8F   1E A2 91 00181 24$:    CMPB    30(R2),#-125                    ; 0792
                                          00V  13 00186          BEQL    26$
                                     59   8F   1E A2 91 00188          CMPB    30(R2),#89
                                          00V  12 0018D          BNEQ    31$
                                          20   27 A2 D1 0018F 26$:    CMPL    39(R2),#32                      ; 0800
                                          00V  19 00193          BLSS    28$
27   A2        7E   8F                     07      00 ED 00195          CMPZV   #0,#7,#^X7E,39(R2)
                                          00V  18 0019C          BGEQ    29$
                                          01   DD 0019E 28$:    PUSHL   #1                                  ; 0806
                                          09   DD 001A0          PUSHL   #9
```

```
                                Generated Code

                                00000000G   EF   9F 001A2          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 001A8          CALLS   #3,PAS$WRITE_CHAR
                                       27     A2   9F 001AF          PUSHAB  39(R2)
                          00000000G  EF        01   FB 001B2          CALLS   #1,NUM_LEN
                                       50     DD 001B9          PUSHL   R0
                                       27     A2   DD 001BB          PUSHL   39(R2)
                                00000000G   EF   9F 001BE          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 001C4          CALLS   #3,PAS$WRITE_INTEGER
                                       00V    11 001CB          BRB     31$
                                FFFFF660    EF   9F 001CD 29$:     PUSHAB  C.ACC                                   ; 0810
                                       02     DD 001D3          PUSHL   #2
                                00000000G   EF   9F 001D5          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 001DB          CALLS   #3,PAS$WRITE_STRING
                                       01     DD 001E2          PUSHL   #1
                                       27     A2   DD 001E4          PUSHL   39(R2)
                                00000000G   EF   9F 001E7          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 001ED          CALLS   #3,PAS$WRITE_CHAR
                                       01     DD 001F4          PUSHL   #1
                                       27     DD 001F6          PUSHL   #39
                                00000000G   EF   9F 001F8          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 001FE          CALLS   #3,PAS$WRITE_CHAR
                                21     1E     A2   91 00205 31$:     CMPB    30(R2),#33                      ; 0817
                                       03     13 00209          BEQL    .+3
                                          0000V  31 0020B          BRW     42$
                                10BF    CF        00   FB 0020E          CALLS   #0,SHOW_QUALIFIER               ; 0821
                            00000100    8F     23   A2   D1 00213          CMPL    35(R2),#256                     ; 0823
                                       00V    1E 0021B          BGEQU   34$
                         00VFFFFF611    EF     23   A2   E1 0021D          BBC     35(R2),C.ACD,34$
                                       01     DD 00226          PUSHL   #1                                      ; 0830
                                       09     DD 00228          PUSHL   #9
                                00000000G   EF   9F 0022A          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 00230          CALLS   #3,PAS$WRITE_CHAR
                                       27     A2   9F 00237          PUSHAB  39(R2)
                          00000000G  EF        01   FB 0023A          CALLS   #1,NUM_LEN
                                       50     DD 00241          PUSHL   R0
                                       27     A2   DD 00243          PUSHL   39(R2)
                                00000000G   EF   9F 00246          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 0024C          CALLS   #3,PAS$WRITE_INTEGER
                                          0000V  31 00253          BRW     42$
                       01               03     23   A2   CF 00256 34$:     CASEL   35(R2),#3,#1                    ; 0836
                                          0000V     0025B          .DISPL  35$
                                          0000V     0025D          .DISPL  36$
                                          0000V  31 0025F          BRW     39$
                                FFFFF5EF    EF   9F 00262 35$:     PUSHAB  C.ACE                                   ; 0842
                                       02     DD 00268          PUSHL   #2
                                00000000G   EF   9F 0026A          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 00270          CALLS   #3,PAS$WRITE_STRING
                                       34     A2   9F 00277          PUSHAB  52(R2)
                          00000000G  EF        01   FB 0027A          CALLS   #1,NUM_LEN
                                       50     DD 00281          PUSHL   R0
                                       34     A2   DD 00283          PUSHL   52(R2)
                                00000000G   EF   9F 00286          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 0028C          CALLS   #3,PAS$WRITE_INTEGER
                                       01     DD 00293          PUSHL   #1
                                       2C     DD 00295          PUSHL   #44
                                00000000G   EF   9F 00297          PUSHAB  FDL_DEST
                          00000000G  EF        03   FB 0029D          CALLS   #3,PAS$WRITE_CHAR
```

```
                          38   A2   9F 002A4          PUSHAB  56(R2)
00000000G   EF                 01   FB 002A7          CALLS   #1,NUM_LEN
                               50   DD 002AE          PUSHL   R0
                          38   A2   DD 002B0          PUSHL   56(R2)
                    00000000G  EF   9F 002B3          PUSHAB  FDL_DEST
00000000G   EF                 03   FB 002B9          CALLS   #3,PAS$WRITE_INTEGER
                               01   DD 002C0          PUSHL   #1
                               2C   DD 002C2          PUSHL   #44
                    00000000G  EF   9F 002C4          PUSHAB  FDL_DEST
00000000G   EF                 03   FB 002CA          CALLS   #3,PAS$WRITE_CHAR
                          3C   A2   9F 002D1          PUSHAB  60(R2)
00000000G   EF                 01   FB 002D4          CALLS   #1,NUM_LEN
                               50   DD 002DB          PUSHL   R0
                          3C   A2   DD 002DD          PUSHL   60(R2)
                    00000000G  EF   9F 002E0          PUSHAB  FDL_DEST
00000000G   EF                 03   FB 002E6          CALLS   #3,PAS$WRITE_INTEGER
                               01   DD 002ED          PUSHL   #1
                               29   DD 002EF          PUSHL   #41
                    00000000G  EF   9F 002F1          PUSHAB  FDL_DEST
00000000G   EF                 03   FB 002F7          CALLS   #3,PAS$WRITE_CHAR
                               00V  11 002FE          BRB     42$
                          11   A2   9F 00300 36$:      PUSHAB  17(R2)                          ; 0853
      0D4C  CF                 01   FB 00303          CALLS   #1,CHECK_QUOTES
                          11   A2   B5 00308          TSTW    17(R2)                          ; 0855
                               00V  15 0030B          BLEQ    42$
                               01   DD 0030D          PUSHL   #1                              ; 0857
                               09   DD 0030F          PUSHL   #9
                    00000000G  EF   9F 00311          PUSHAB  FDL_DEST
00000000G   EF                 03   FB 00317          CALLS   #3,PAS$WRITE_CHAR
                    7E    11   A2   3C 0031E          MOVZWL  17(R2),-(SP)
                               00   DD 00322          PUSHL   #0
                          15   B2   9F 00324          PUSHAB  @21(R2)
                    000000FF   8F   DD 00327          PUSHL   #255
                    00000000G  EF   9F 0032D          PUSHAB  FDL_DEST
00000000G   EF                 05   FB 00333          CALLS   #5,PAS$WRITE_STRING
                               00V  11 0033A          BRB     42$
                                    0033C 39$:
      63    8F         1E    A2   91 0033C 42$:      CMPB    30(R2),#99                      ; 0876
                               00V  12 00341          BNEQ    45$
00000000G   EF           2C    A2   D0 00343          MOVL    44(R2),TEMP_INT2                ; 0880
                    00000000G  EF   9F 0034B          PUSHAB  TEMP_INT2                       ; 0881
      F4    AD   010E00FF      8F   D0 00351          MOVL    #17694975,-12(FP)
      F8    AD   00000000G     EF   9E 00359          MOVAB   TEMP_STRING255,-8(FP)
                         F4    AD   9F 00361          PUSHAB  -12(FP)
                         FC    AD   9F 00364          PUSHAB  RETLEN
      EC    AD   010E0005      8F   D0 00367          MOVL    #17694725,-20(FP)
      F0    AD   FFFFF4E4      EF   9E 0036F          MOVAB   C_ACF,-16(FP)
                         EC    AD   9F 00377          PUSHAB  -20(FP)
00000000G   EF                 04   FB 0037A          CALLS   #4,SYS$FAOL
                    7E    FC   AD   32 00381          CVTWL   RETLEN,-(SP)                    ; 0882
                               00   DD 00385          PUSHL   #0
                    00000000G  EF   9F 00387          PUSHAB  TEMP_STRING255
                    000000FF   8F   DD 0038D          PUSHL   #255
                    00000000G  EF   9F 00393          PUSHAB  FDL_DEST
00000000G   EF                 05   FB 00399          CALLS   #5,PAS$WRITE_STRING
      65    8F         1E    A2   91 003A0 45$:      CMPB    30(R2),#101                     ; 0889
                               03   13 003A5          BEQL    .+3
```

```
                    0000V  31 003A7          BRW     47$
            FFFFF4B3   EF  9F 003AA          PUSHAB  C.ACG                                   ; 0893
                       09  DD 003B0          PUSHL   #9
            00000000G  EF  9F 003B2          PUSHAB  FDL_DEST
00000000G   EF             03 003B8          CALLS   #3,PAS$WRITE_STRING
            00000000   8F  DF 003BF          PUSHAL  #0                                      ; 0894
                  30   A2  9F 003C5          PUSHAB  48(R2)
    0000V   CF             02 003C8          CALLS   #2,SHOW_PROT
            FFFFF49C   EF  9F 003CD          PUSHAB  C.ACH                                   ; 0895
                       08  DD 003D3          PUSHL   #8
            00000000G  EF  9F 003D5          PUSHAB  FDL_DEST
00000000G   EF             03 003DB          CALLS   #3,PAS$WRITE_STRING
            00000004   8F  DF 003E2          PUSHAL  #4                                      ; 0896
                  30   A2  9F 003E8          PUSHAB  48(R2)
    0000V   CF             02 003EB          CALLS   #2,SHOW_PROT
            FFFFF481   EF  9F 003F0          PUSHAB  C.ACI                                   ; 0897
                       08  DD 003F6          PUSHL   #8
            00000000G  EF  9F 003F8          PUSHAB  FDL_DEST
00000000G   EF             03 003FE          CALLS   #3,PAS$WRITE_STRING
            00000008   8F  DF 00405          PUSHAL  #8                                      ; 0898
                  30   A2  9F 0040B          PUSHAB  48(R2)
    0000V   CF             02 0040E          CALLS   #2,SHOW_PROT
            FFFFF466   EF  9F 00413          PUSHAB  C.ACJ                                   ; 0899
                       08  DD 00419          PUSHL   #8
            00000000G  EF  9F 0041B          PUSHAB  FDL_DEST
00000000G   EF             03 00421          CALLS   #3,PAS$WRITE_STRING
            0000000C   8F  DF 00428          PUSHAL  #12                                     ; 0900
                  30   A2  9F 0042E          PUSHAB  48(R2)
    0000V   CF             02 00431          CALLS   #2,SHOW_PROT
                       01  DD 00436          PUSHL   #1                                      ; 0901
                       29  DD 00438          PUSHL   #41
            00000000G  EF  9F 0043A          PUSHAB  FDL_DEST
00000000G   EF             03 00440          CALLS   #3,PAS$WRITE_CHAR
                  09   A2  B5 00447  47$:     TSTW    9(R2)                                   ; 0908
                       00V 15 0044A          BLEQ    49$
                       01  DD 0044C          PUSHL   #1                                      ; 0910
                       09  DD 0044E          PUSHL   #9
            00000000G  EF  9F 00450          PUSHAB  FDL_DEST
00000000G   EF             03 00456          CALLS   #3,PAS$WRITE_CHAR
            7E     09  A2  3C 0045D          MOVZWL  9(R2),-(SP)
                       00  DD 00461          PUSHL   #0
                  0D   B2  9F 00463          PUSHAB  @13(R2)
            000000FF   8F  DD 00466          PUSHL   #255
            00000000G  EF  9F 0046C          PUSHAB  FDL_DEST
00000000G   EF             05 00472          CALLS   #5,PAS$WRITE_STRING
            00000000G  EF  9F 00479  49$:     PUSHAB  FDL_DEST                                ; 0916
00000000G   EF             01 0047F          CALLS   #1,PAS$WRITELN2
                       00V 11 00486  50$:     BRB     56$
            50 00000000G   EF  D0 00488  51$:     MOVL    DEF_CURRENT,R0                   ; 0924
                  09   A0  B5 0048F          TSTW    9(R0)                                   ; 0931
                       00V 15 00492          BLEQ    56$
            7E     09  A0  3C 00494          MOVZWL  9(R0),-(SP)                             ; 0933
                       00  DD 00498          PUSHL   #0
                  0D   B0  9F 0049A          PUSHAB  @13(R0)
            000000FF   8F  DD 0049D          PUSHL   #255
            00000000G  EF  9F 004A3          PUSHAB  FDL_DEST
00000000G   EF             05 004A9          CALLS   #5,PAS$WRITE_STRING
```

```
                          00000000G   EF    9F 004B0          PUSHAB  FDL_DEST
                00000000G EF          01    FB 004B6          CALLS   #1,PAS$WRITELN2
                                      00V   11 004BD          BRB     56$
                                            004BF 55$:
                          00000000G   EF    D6 004BF 56$:     INCL    LINES_SHOWN                                     ; 0947
             03 00000000G EF          00    E0 004C5          BBS     #0,DEST_IS_TERMINAL,.+3                         ; 0949
                                    0000V   31 004CD          BRW     66$
                       50 00000000G EF    D0 004D0          MOVL    DEF_CURRENT,R0                                  ; 0953
                             01     A0    D5 004D7          TSTL    1(R0)
                                      00V   13 004DA          BEQL    63$
             50 00000000G EF 00000000G EF    C3 004DC          SUBL3   LINES_SHOWN,LINES_PER_PAGE,R0                   ; 0955
                       50            03    C2 004E8          SUBL2   #3,R0
                       04            50    D1 004EB          CMPL    R0,#4
                                      00V   18 004EE          BGEQ    63$
                       50 00000000G EF    D0 004F0          MOVL    DEF_CURRENT,R0
                       52 00000000G EF    D0 004F7          MOVL    DEF_CURRENT,R2
                       52       01   A2    D0 004FE          MOVL    1(R2),R2
                    19 A2       19   A0    91 00502          CMPB    25(R0),25(R2)
                                      00V   12 00507          BNEQ    61$
                       52 00000000G EF    D0 00509          MOVL    DEF_CURRENT,R2
                       50 00000000G EF    D0 00510          MOVL    DEF_CURRENT,R0
                       50       01   A0    D0 00517          MOVL    1(R0),R0
                    1A A0       1A   A2    D1 0051B          CMPL    26(R2),26(R0)
                                      00V   13 00520          BEQL    63$
                          00000000G   EF    D4 00522 61$:     CLRL    LINES_SHOWN                                     ; 0969
                          00000002    8F    DF 00528          PUSHAL  #2                                              ; 0970
                00000000G EF          01    FB 0052E          CALLS   #1,CLEAR
             50 00000000G EF          03    C3 00535 63$:     SUBL3   #3,LINES_PER_PAGE,R0                            ; 0974
                       50 00000000G EF    D1 0053D          CMPL    LINES_SHOWN,R0
                                      00V   19 00544          BLSS    66$
                          00000000G   EF    D4 00546          CLRL    LINES_SHOWN                                     ; 0984
                          00000002    8F    DF 0054C          PUSHAL  #2                                              ; 0985
                   00000000G EF           01    FB 00552          CALLS   #1,CLEAR
                                               04 00559 66$:     RET                                                     ; 0991

; Routine Size: 1370 bytes,    Routine Base: $CODE + 014C5

                                            00000 SHOW_PROT:                                                          ; 0661
                                      0004  00000          .WORD   ^M<R2>
                       52       04   BC    D0 00002          MOVL    @4(R12),PROTECTION
                       5C       08   BC    D0 00006          MOVL    @8(R12),FIELD_OFFSET
                00V    52            5C    E1 0000A          BBC     FIELD_OFFSET,PROTECTION,2$                      ; 0665
                                      01    DD 0000E          PUSHL   #1                                              ; 0667
                       7E       52   8F    9A 00010          MOVZBL  #82,-(SP)
                          00000000G   EF    9F 00014          PUSHAB  FDL_DEST
                   00000000G EF           03    FB 0001A          CALLS   #3,PAS$WRITE_CHAR
                       50       01   AC    9E 00021 2$:      MOVAB   1(FIELD_OFFSET),R0                              ; 0669
                00V    52            50    E1 00025          BBC     R0,PROTECTION,4$
                                      01    DD 00029          PUSHL   #1                                              ; 0671
                       7E       57   8F    9A 0002B          MOVZBL  #87,-(SP)
                          00000000G   EF    9F 0002F          PUSHAB  FDL_DEST
                   00000000G EF           03    FB 00035          CALLS   #3,PAS$WRITE_CHAR
                       50       02   AC    9E 0003C 4$:      MOVAB   2(FIELD_OFFSET),R0                              ; 0673
                00V    52            50    E1 00040          BBC     R0,PROTECTION,6$
                                      01    DD 00044          PUSHL   #1                                              ; 0675
                       7E       45   8F    9A 00046          MOVZBL  #69,-(SP)
                          00000000G   EF    9F 0004A          PUSHAB  FDL_DEST
```

```
                    00000000G  EF                03   FB 00050              CALLS   #3,PAS$WRITE_CHAR
                               5C          03     AC   9E 00057 6$:         MOVAB   3(FIELD_OFFSET),R12                     ; 0677
                         00V   52                 5C   E1 0005B              BBC     R12,PROTECTION,8$
                                                  01   DD 0005F              PUSHL   #1                                      ; 0679
                               7E          44     8F   9A 00061              MOVZBL  #68,-(SP)
                                     00000000G    EF   9F 00065              PUSHAB  FDL_DEST
                    00000000G  EF                03   FB 0006B              CALLS   #3,PAS$WRITE_CHAR
                                                  04   00072 8$:            RET                                             ; 0681

; Routine Size: 115 bytes,    Routine Base: $CODE + 01A1F

                                                       00000 GENERATE_FDL:                                                  ; 1041
                                                  0004 00000                .WORD   ^M<R2>
                    00000000G  EF  00000000G     EF    D0 00002              MOVL    DEF_HEAD,DEF_CURRENT                    ; 1052
                               5C                00    D2 0000D              MCOML   #0,PREV_PRINUM                          ; 1057
                                     00000000G   EF    D4 00010              CLRL    LINES_SHOWN                             ; 1062
                                     00000000G   EF    D5 00016              TSTL    DEF_CURRENT                             ; 1067
                                                 00V   13 0001C              BEQL    11$
                               50 00000000G      EF    D0 0001E 2$:          MOVL    DEF_CURRENT,R0                          ; 1082
                                                 5C    D5 00025              TSTL    PREV_PRINUM                             ; 1086
                                                 00V   18 00027              BGEQ    5$
                               52          19    A0    90 00029              MOVB    25(R0),PREV_PRIMARY                     ; 1093
                               5C          1A    A0    D0 0002D              MOVL    26(R0),PREV_PRINUM                      ; 1094
                                                 00V   11 00031              BRB     9$
                         1A    A0                5C    D1 00033 5$:          CMPL    PREV_PRINUM,26(R0)                      ; 1105
                                                 00V   12 00037              BNEQ    7$
                         19    A0                52    91 00039              CMPB    PREV_PRIMARY,25(R0)
                                                 00V   13 0003D              BEQL    9$
                               52          19    A0    90 0003F 7$:          MOVB    25(R0),PREV_PRIMARY                     ; 1111
                               5C          1A    A0    D0 00043              MOVL    26(R0),PREV_PRINUM                      ; 1112
                                     00000000G   EF    9F 00047              PUSHAB  FDL_DEST                                ; 1117
                    00000000G  EF                01    FB 0004D              CALLS   #1,PAS$WRITELN2
                                     00000000G   EF    D6 00054              INCL    LINES_SHOWN                             ; 1118
                                           01    8F    9F 0005A 9$:          PUSHAB  #1                                      ; 1126
                       14C5    CF                01    FB 0005D              CALLS   #1,SHOW_CURRENT
                    00000000G  EF                00    FB 00062              CALLS   #0,INCR_CURRENT                         ; 1127
                                     00000000G   EF    D5 00069              TSTL    DEF_CURRENT
                                                 AD    12 0006F              BNEQ    2$
                                                       04 00071 11$:         RET                                             ; 1133

; Routine Size: 114 bytes,    Routine Base: $CODE + 01A92

                                                       00000 VIEW_DEF:                                                      ; 1183
                                                  0000 00000                .WORD   ^M<>
                                     00000003    8F    DF 00002              PUSHAL  #3                                      ; 1190
                    00000000G  EF                01    FB 00008              CALLS   #1,CLEAR
                    00000000G  EF                01    90 0000F              MOVB    #1,DEST_IS_TERMINAL                     ; 1196
                                           00V   AF    9F 00016              PUSHAB  1$                                      ; 1198
                                                 19    DD 00019              PUSHL   #25
                                     00000000G   EF    9F 0001B              PUSHAB  FDL_DEST
                    00000000G  EF                03    FB 00021              CALLS   #3,PAS$CLOSE2
                                     000000FC    8F    DD 00028 1$:          PUSHL   #252                                    ; 1199
                                                 07    DD 0002E              PUSHL   #7
                                                 04    DD 00030              PUSHL   #4
                                     00000000G   EF    9F 00032              PUSHAB  SYS$OUTPUT_NAME
                                                 0B    DD 00038              PUSHL   #11
                                                 01    DD 0003A              PUSHL   #1
```

```
                                 00000000G   EF   9F 0003C                PUSHAB  FDL_DEST
                       00000000G EF          07   FB 00042                CALLS   #7,PAS$OPEN2
                                 00000000G   EF   9F 00049                PUSHAB  FDL_DEST                        ; 1200
                       00000000G EF          01   FB 0004F                CALLS   #1,PAS$REWRITE2
                          1A92   CF          00   FB 00056                CALLS   #0,GENERATE_FDL                 ; 1205
                                 00000000G   EF   9F 0005B                PUSHAB  FDL_DEST                        ; 1210
                       00000000G EF          01   FB 00061                CALLS   #1,PAS$CLOSE2
                    00V00000000G EF          00   E0 00068                BBS     #0,CONTROL_ZEE_TYPED,4$         ; 1215
                                 00000000G   EF   D5 00070                TSTL    LINES_SHOWN
                                             00V  13 00076                BEQL    4$
                                 00000000G   EF   9F 00078                PUSHAB  PAS$FV_OUTPUT                   ; 1219
                       00000000G EF          01   FB 0007E                CALLS   #1,PAS$WRITELN2
                                 00000002    8F   DF 00085                PUSHAL  #2                              ; 1220
                       00000000G EF          01   FB 0008B                CALLS   #1,CLEAR
                                             04   00092 4$:              RET                                     ; 1224

; Routine Size: 147 bytes,    Routine Base: $CODE + 01B04

                                             003C 00000                .ENTRY  SHOW_PRIMARY_SECTION,^M<R2,R3,R4,R5>    ; 1276
                                 5E          C0   AE 00002             MOVAB   -64(SP),SP
                    C0   AD         04       BC   8F 00006             MOVC3   #64,@4(R12),TEST
                                 00000000G   EF   D4 0000E             CLRL    LINES_SHOWN                     ; 1286
                       00000000G EF          00000000G EF   D0 00014   MOVL    DEF_HEAD,DEF_CURRENT            ; 1291
                                             5C   94 0001F             CLRB    AT_PRIMARY                      ; 1292
                                          00 8F   9F 00021 1$:         PUSHAB  #0                              ; 1299
                                          C0 AD   9F 00024             PUSHAB  TEST
                       00000000G EF          02   FB 00027             CALLS   #2,CURRENT_EQ_TEST
                                 00V         50   E9 0002E             BLBC    R0,3$
                                          01 8F   9F 00031             PUSHAB  #1                              ; 1303
                          14C5   CF          01   FB 00034             CALLS   #1,SHOW_CURRENT
                                 5C          01   90 00039             MOVB    #1,AT_PRIMARY                   ; 1304
                                             00V  11 0003C             BRB     6$
                                 00V         5C   E9 0003E 3$:         BLBC    AT_PRIMARY,6$                   ; 1312
                                 00000000G   EF   D4 00041             CLRL    DEF_CURRENT                     ; 1314
                       00000000G EF          00   FB 00047 6$:         CALLS   #0,INCR_CURRENT                 ; 1321
                                 00000000G   EF   D5 0004E             TSTL    DEF_CURRENT
                                             CB   12 00054             BNEQ    1$
                                 00000000G   EF   9F 00056             PUSHAB  FDL_DEST                        ; 1325
                       00000000G EF          01   FB 0005C             CALLS   #1,PAS$WRITELN2
                                             04   00063               RET                                     ; 1327

; Routine Size: 100 bytes,    Routine Base: $CODE + 01B97

                                             0000 00000                .ENTRY  SHOW_ALL_PRIMARIES,^M<>         ; 1379
                                 00000000G   EF   D4 00002             CLRL    LINES_SHOWN                     ; 1386
                       00000000G EF          00000000G EF   D0 00008   MOVL    DEF_HEAD,DEF_CURRENT            ; 1391
                                 50          00000000G EF   D0 00013 1$:   MOVL    DEF_CURRENT,R0                  ; 1399
                                             60   95 0001A             TSTB    (R0)
                                             00V  12 0001C             BNEQ    4$
                                 50          00000000G EF   D0 0001E   MOVL    DEF_CURRENT,R0
                                 09             19     A0   91 00025   CMPB    25(R0),#9
                                             00V  13 00029             BEQL    4$
                                          01 8F   9F 0002B             PUSHAB  #1                              ; 1405
                          14C5   CF          01   FB 0002E             CALLS   #1,SHOW_CURRENT
                       00000000G EF          00   FB 00033 4$:         CALLS   #0,INCR_CURRENT                 ; 1410
                                 00000000G   EF   D5 0003A             TSTL    DEF_CURRENT
                                             D1   12 00040             BNEQ    1$
```

```
                          00000000G   EF    9F 00042          PUSHAB  FDL_DEST                                        ; 1414
                00000000G EF          01    FB 00048          CALLS   #1,PAS$WRITELN2
                                            04 0004F          RET                                                     ; 1416

; Routine Size: 80 bytes,    Routine Base: $CODE + 01BFB

                                            0004 00000        .ENTRY  SHOW_CUR_PRI_SEC,^M<R2>                         ; 1468
                          5E                04    C2 00002    SUBL2   #4,SP
                          5C        04      BC    90 00005    MOVB    @4(R12),PLUS_VALUE
                    00000000G             EF    D4 00009    CLRL    LINES_SHOWN                                     ; 1475
                          50  00000000G     EF    D0 0000F    MOVL    DEF_CURRENT,R0                                  ; 1482
                          52                60    90 00016    MOVB    (R0),SAVE_OBJECT_TYPE
                          50  00000000G     EF    D0 00019    MOVL    DEF_CURRENT,R0                                  ; 1484
                                            60    94 00020    CLRB    (R0)
                       FC     AD            5C    90 00022    MOVB    PLUS_VALUE,-4(FP)                               ; 1485
                                 FC         AD    9F 00026    PUSHAB  -4(FP)
                      14C5     CF           01    FB 00029    CALLS   #1,SHOW_CURRENT
                          50  00000000G     EF    D0 0002E    MOVL    DEF_CURRENT,R0                                  ; 1487
                          0F        19      A0    91 00035    CMPB    25(R0),#15
                                            00V   13 00039    BEQL    2$
                          50  00000000G     EF    D0 0003B    MOVL    DEF_CURRENT,R0                                  ; 1491
                          60                01    90 00042    MOVB    #1,(R0)
                       FC     AD            5C    90 00045    MOVB    PLUS_VALUE,-4(FP)                               ; 1492
                                 FC         AD    9F 00049    PUSHAB  -4(FP)
                      14C5     CF           01    FB 0004C    CALLS   #1,SHOW_CURRENT
                          50  00000000G     EF    D0 00051 2$:MOVL    DEF_CURRENT,R0                                  ; 1496
                          60                52    90 00058    MOVB    SAVE_OBJECT_TYPE,(R0)
                                            04 0005B          RET                                                     ; 1498

; Routine Size: 92 bytes,    Routine Base: $CODE + 01C4B

                                            01CA7             .END
```

```
COMMAND QUALIFIERS

  PASCAL/MACHINE/NODEBUG/NOCHECK/LIS=LIS$:EDFSHOW/OBJ=OBJ$:EDFSHOW MSRC$:EDFSHOW

  /CHECK=(NOBOUNDS,NOCASE_SELECTORS,NOOVERFLOW,NOPOINTERS,NOSUBRANGE)
  /DEBUG=(NOSYMBOLS,NOTRACEBACK)
  /ENVIRONMENT= $255$DUA28:[EDF.OBJ]EDFSHOW.PEN;1
  /LIST= $255$DUA28:[EDF.LIS]EDFSHOW.LIS;1
  /OBJECT= $255$DUA28:[EDF.OBJ]EDFSHOW.OBJ;1
  /NOCROSS_REFERENCE /ERROR_LIMIT=30 /NOG_FLOATING /MACHINE_CODE /NOOLD_VERSION /OPTIMIZE /NOSTANDARD /WARNINGS


COMPILER INTERNAL TIMING

  Phase                  Faults      CPU Time      Elapsed Time
  Initialization             83      00:00.4          00:02.5
  Source Analysis           675      00:15.7          03:51.6
  Source Listing             41      00:01.9          00:04.0
  Tree Construction         185      00:01.0          00:02.1
  Flow Analysis              39      00:00.3          00:00.5
  Profit Analysis            30      00:00.5          00:01.1
  Context Analysis          693      00:07.8          00:17.1
  Name Packing               10      00:00.2          00:00.5
  Code Selection             63      00:01.1          00:02.7
  Final                     361      00:06.2          00:16.8
  TOTAL                    2184      00:35.3          04:39.2


COMPILATION STATISTICS

  CPU Time:        00:35.3        (2554 Lines/Minute)
  Elapsed Time:    04:39.2
  Page Faults:     2184
  Compilation Complete
```

EDFSDLMSG
LIS

EDFMAIN
LIS

EDFSHOW
LIS

EDFGRF
LIS

EDFMSG
LIS

EDFTABLES
LIS

EDFTYPE
LIS

EDFUTIL
LIS

EDFTERM
LIS

EDFSTRUCT
LIS