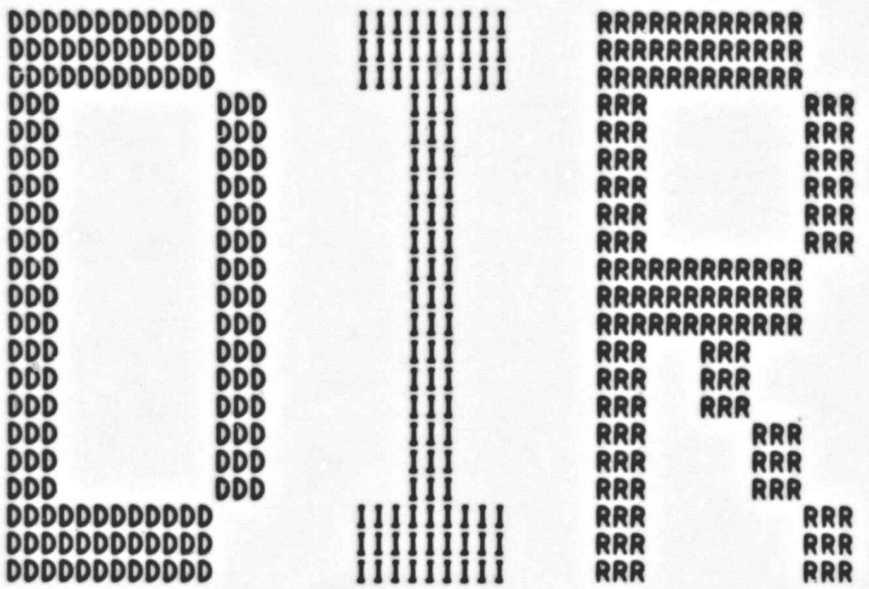
DDDDDDDDDDD IIIIIIIII RRRRRRRRRRR
DDDDDDDDDDDD IIIIIIIII RRRRRRRRRRRR
DDDDDDDDDDDD IIIIIIIII RRRRRRRRRRRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRRRRRRRRRR
DDD DDD III RRRRRRRRRRRR
DDD DDD III RRRRRRRRRRRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDD DDD III RRR RRR
DDDDDDDDDDDD IIIIIIIII RRR RRR
DDDDDDDDDDDD IIIIIIIII RRR RRR
DDDDDDDDDDD IIIIIIIII RRR RRR

FILEID**DISPLYDEF

```
DDDDDDDD     IIIIII     SSSSSSSS  PPPPPPPP   LL          YY      YY  DDDDDDDD   EEEEEEEEEE  FFFFFFFFFF
DDDDDDDD     IIIIII     SSSSSSSS  PPPPPPPP   LL          YY      YY  DDDDDDDD   EEEEEEEEEE  FFFFFFFFFF
DD      DD     II     SS        PP      PP  LL          YY      YY  DD      DD  EE          FF
DD      DD     II     SS        PP      PP  LL          YY      YY  DD      DD  EE          FF
DD      DD     II     SS        PP      PP  LL            YY  YY    DD      DD  EE          FF
DD      DD     II     SS        PP      PP  LL            YY  YY    DD      DD  EE          FF
DD      DD     II       SSSSSS  PPPPPPPP   LL              YY      DD      DD  EEEEEEEE    FFFFFFFF
DD      DD     II       SSSSSS  PPPPPPPP   LL              YY      DD      DD  EEEEEEEE    FFFFFFFF
DD      DD     II             SS PP         LL              YY      DD      DD  EE          FF
DD      DD     II             SS PP         LL              YY      DD      DD  EE          FF
DD      DD     II             SS PP         LL              YY      DD      DD  EE          FF          ....
DD      DD     II             SS PP         LL              YY      DD      DD  EE          FF          ....
DDDDDDDD     IIIIII     SSSSSSSS  PP         LLLLLLLLLL      YY      DDDDDDDD   EEEEEEEEEE  FF          ....
DDDDDDDD     IIIIII     SSSSSSSS  PP         LLLLLLLLLL      YY      DDDDDDDD   EEEEEEEEEE  FF          ....
```

```
  SSSSSSSS  DDDDDDDD    LL
  SSSSSSSS  DDDDDDDD    LL
SS          DD      DD  LL
SS          DD      DD  LL
SS          DD      DD  LL
SS          DD      DD  LL
  SSSSSS    DD      DD  LL
  SSSSSS    DD      DD  LL
        SS  DD      DD  LL
        SS  DD      DD  LL
        SS  DD      DD  LL
        SS  DD      DD  LL
SSSSSSSS    DDDDDDDD    LLLLLLLLLL
SSSSSSSS    DDDDDDDD    LLLLLLLLLL
```

```
{
{       IDENT = 'V04-000'
{
{*****************************************************************************
{*                                                                           *
{*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
{*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
{*  ALL RIGHTS RESERVED.                                                     *
{*                                                                           *
{*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
{*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
{*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
{*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
{*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
{*  TRANSFERRED.                                                             *
{*                                                                           *
{*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
{*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
{*  CORPORATION.                                                             *
{*                                                                           *
{*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
{*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
{*                                                                           *
{*****************************************************************************

{++
{
{ FACILITY:     DIRECTORY
{
{ ABSTRACT:
{
{       This module contains the definition for the display information
{       block.  This is the block that contains all of the information
{       that may be displayed about a file.
{
{ ENVIRONMENT:
{
{       VAX/VMS operating system, unprivileged user mode utilities
{--
{
{ AUTHOR:       L. Mark Pilant          CREATION DATE:  4-Mar-1983
{
{ MODIFIED BY:
{
{       V03-007 LMP0211         L. Mark Pilant,         10-Mar-1984  12:47
{               Modify the display block to more accurately track the
{               information now obtained directly from the disk ACP.
{
{       V03-006 LMP0182         L. Mark Pilant,         11-Jan-1984  12:38
{               Add flags to indicate what selection option was chosen.
{
{       V03-005 DAS0002         David Solomon           01-Aug-1983
{               Max journal name went from 12 to 16.
{
```

```
{       V03-004 DAS0001         David Solomon           29-Jul-1983
{               XAB$M_RUA is now XAB$M_ONLY_RU.
{
{       V03-003 LMP0119         L. Mark Pilant,         15-Jun-1983  9:26
{               Add support for identifiers.
{
{       V03-002 LMP0108         L. Mark Pilant,         28-Apr-1983  12:21
{               Add support for RMS journal names.
{
{       V03-001 LMP0100         L. Mark Pilant,         14-Apr-1983  11:50
{               Misc fixups.
{
{**
```

```
module  $DISPLAYDEF;

/*
/* Define the display information block.  This block is filled with all of
/* the information needed to create a full directory display on the selected
/* file.  With the exception of network files and indexed files, all of the
/* information about the file is obtained from the disk ACP.  Network files
/* and indexed files are handled by RMS.  Because the primary path is the disk
/* ACP, the display block should parallel the ODS-2 file header structurs as
/* closely as possible.
/*

aggregate DISPLAYDEF structure prefix DIR_;
{
{ The following describe the file specification
{
    STATUS longword unsigned;                   /* File OPEN/QIO status
    FLAGS structure longword unsigned;          /* Miscellaneous flags
        NOPRIV bitfield mask;                   /* No privilege to read info
        SQD bitfield mask;                      /* True if seq. device (magtapes)
    end FLAGS;
    DVI character dimension 16;                 /* Device ident.
    FNS byte unsigned;                          /* File name size
    FILENAME character dimension 256;           /* File name buffer
    NODE byte unsigned;                         /* Filespec nodename length
    DEV byte unsigned;                          /* Filespec device length
    DIR byte unsigned;                          /* Filespec directory length
    VER byte unsigned;                          /* Filespec version length
    VERLIMIT word unsigned;                     /* File version limit
    ACLLENGTH longword unsigned;                /* Size of file's ACL
{
{ The following are obtained from the file header ($FHxDEF)
{
    FID structure fill;                         /* file ID
        FID_NUM word unsigned;                  /* file number
        FID_SEQ word unsigned;                  /* file sequence number
        FID_RVN word unsigned;                  /* relative volume number
    end FID;
    RECATTR structure;                          /* file record attributes
        RTYPE structure byte unsigned;          /* record type
            RTYPE bitfield length 4;            /* record type subfield
            constant(
                  UNDEFINED                     /* undefined record type
                , FIXED                         /* fixed record type
                , "VARIABLE"                    /* variable length
                , VFC                           /* variable + fixed control
                , STREAM                        /* RMS-11 (DEC traditional) stream format
                , STREAMLF                      /* LF-terminated stream format
                , STREAMCR                      /* CR-terminated stream format
                ) equals 0 increment 1  prefix DIR_ tag C;
            FILEORG bitfield length 4;          /* file organization
            constant(
                  SEQUENTIAL                    /* sequential organization
                , RELATIVE                      /* relative organization
                , INDEXED                       /* indexed organization
                , DIRECT                        /* direct organization
```

```
            ) equals 0 increment 1  prefix DIR_ tag C;
        end RTYPE;
        RATTRIB structure byte unsigned;        /* record attributes
            FORTRANCC bitfield mask;            /* Fortran carriage control
            IMPLIEDCC bitfield mask;            /* implied carriage control
            PRINTCC bitfield mask;              /* print file carriage control
            NOSPAN bitfield mask;               /* no spanned records
        end RATTRIB;
        RSIZE word unsigned;                    /* record size in bytes
        HIBLK structure longword unsigned;      /* highest allocated VBN
            HIBLKH word unsigned;               /* high order word
            HIBLKL word unsigned;               /* low order word
        end HIBLK;
        EFBLK structure longword unsigned;      /* end of file VBN
            EFBLKH word unsigned;               /* high order word
            EFBLKL word unsigned;               /* low order word
        end EFBLK;
        FFBYTE word unsigned;                   /* first free byte in EFBLK
        BKTSIZE byte unsigned;                  /* bucket size in blocks
        VFCSIZE byte unsigned;                  /* size in bytes of fixed length control for VFC records
        MAXREC word unsigned;                   /* maximum record size in bytes
        DEFEXT word unsigned;                   /* default extend quantity
        GBC word unsigned;                      /* global buffer count
        FILL_1 word dimension 4 fill;           /* spare
        VERSIONS word unsigned;                 /* default version limit for directory file
    end RECATTR;
    FILECHAR structure longword unsigned;       /* file characteristics
        FILL_1 bitfield fill;                   /* reserved
        NOBACKUP bitfield mask;                 /* file is not to be backed up
        WRITEBACK bitfield mask;                /* file may be write-back cached
        READCHECK bitfield mask;                /* verify all read operations
        WRITCHECK bitfield mask;                /* verify all write operations
        CONTIGB bitfield mask;                  /* keep file as contiguous as possible
        LOCKED bitfield mask;                   /* file is deaccess locked
        CONTIG bitfield mask;                   /* file is contiguous
        FILL_2 bitfield length 3 fill;          /* reserved
        BADACL bitfield mask;                   /* ACL is invalid
        SPOOL bitfield mask;                    /* intermediate spool file
        DIRECTORY bitfield mask;                /* file is a directory
        BADBLOCK bitfield mask;                 /* file contains bad blocks
        MARKDEL bitfield mask;                  /* file is marked for delete
        NOCHARGE bitfield mask;                 /* file space is not to be charged
        ERASE bitfield mask;                    /* erase file contents before deletion
    end FILECHAR;
    ACC_MODE byte unsigned;                     /* least privileged access mode
    FILEOWNER structure longword unsigned;      /* file owner UIC
        UICMEMBER word unsigned;                /* UIC member number
        UICGROUP word unsigned;                 /* UIC group number
    end UIC;
    FILEPROT word unsigned;                     /* file protection
    JOURNAL structure word unsigned;
        ONLY_RU bitfield mask;                  /* file is accessible only in recovery unit
        RUJNL bitfield mask;                    /* enable recovery unit journal
        BIJNL bitfield mask;                    /* enable before image journal
        AIJNL bitfield mask;                    /* enable after image journal
        ATJNL bitfield mask;                    /* enable audit trail journal
```

```
        NEVER_RU bitfield mask;                 /* file is never accessible in recovery unit
    end JOURNAL;
    HIGHWATER longword unsigned;                /* high-water mark in file
    CLASS_PROT structure;                       /* security classification mask
        FILL_5 byte dimension 20 fill;          /* see structure in $CLSDEF
        end CLASS_PROT;
{
{ The following are obtained from the file header ident area ($FIxDEF)
{
    REVISION word unsigned;                     /* revision number (binary)
    CREDATE structure quadword unsigned;        /* Creation date
        CDT0 longword unsigned;
        CDT4 longword unsigned;
    end CREDATE;
    REVDATE structure quadword unsigned;        /* Revision date
        RDT0 longword unsigned;
        RDT4 longword unsigned;
    end REVDATE;
    EXPDATE structure quadword unsigned;        /* Expiration date
        EDT0 longword unsigned;
        EDT4 longword unsigned;
    end EXPDATE;
    BAKDATE structure quadword unsigned;        /* Backup date
        BDT0 longword unsigned;
        BDT4 longword unsigned;
    end BAKDATE;
{
{ The following are obtained from RMS for indexed or relative files.
{
    MRN longword unsigned;                      /* Maximum record number
    NOAREAS byte unsigned;                      /* Number of areas in idx files
    NOKEYS byte unsigned;                       /* Number of keys (ISAM only)
    PVN word unsigned;                          /* Prologue version number
{
{ The following are obtained from the file's ACL.
{
    JNL1 union fill;
        AI_JNLNAME character dimension 17;      /* AI journal name
        JN[2 structure fill;
            AI_SIZE byte unsigned;
            AI_NAME character dimension 16;
        end JN[2;
    end JNL1;
    JNL3 union fill;
        BI_JNLNAME character dimension 17;      /* BI journal name
        JN[4 structure fill;
            BI_SIZE byte unsigned;
            BI_NAME character dimension 16;
        end JN[4;
    end JNL3;
    JNL5 union fill;
        AT_JNLNAME character dimension 17;      /* AT journal name
        JN[6 structure fill;
            AT_SIZE byte unsigned;
            AT_NAME character dimension 16;
        end JN[6;
```

```
    end JNL5;

    constant "LENGTH" equals . prefix DIR_ tag K;
    constant "LENGTH" equals . prefix DIR_ tag C;

end DISPLAYDEF;

end_module $DISPLAYDEF;
```

```
module $DIRQUALDEF;

/*
/* Define the flags needed to determine what qualifiers were given on the
/* command line.
/*

aggregate QUALDEF union prefix DIR_;
QUAL_BITS structure fill;
    QUAL_ACL bitfield;                  /* /ACLS
    QUAL_BRIE bitfield;                 /* /BRIEF
    QUAL_COLU bitfield;                 /* /COLUMN
    QUAL_DATE bitfield;                 /* /DATE
    DATE_CRE bitfield;                  /* /DATE=CREATED
    DATE_EXP bitfield;                  /* /DATE=EXPIRED
    DATE_MOD bitfield;                  /* /DATE=MODIFIED
    DATE_BAK bitfield;                  /* /DATE=BACKUP
    QUAL_FID bitfield;                  /* /FILE_ID
    QUAL_FULL bitfield;                 /* /FULL
    QUAL_GRAN bitfield;                 /* /GRAND_TOTAL
    QUAL_HEAD bitfield;                 /* /HEADING
    QUAL_OUTP bitfield;                 /* /OUTPUT
    QUAL_OWNE bitfield;                 /* /OWNER
    QUAL_PRIN bitfield;                 /* /PRINTER
    QUAL_PROT bitfield;                 /* /PROTECTION
    QUAL_SECU bitfield;                 /* /SECURITY
    SELE_ACL bitfield;                  /* /SELECT=ACL
    SELE_SIZE bitfield;                 /* /SELECT=SIZE
    QUAL_SIZE bitfield;                 /* /SIZE
    SIZE_ALL bitfield;                  /* /SIZE=ALL
    SIZE_ALLO bitfield;                 /* /SIZE=ALLOCATION
    SIZE_USED bitfield;                 /* /SIZE=USED
    QUAL_TOTL bitfield;                 /* /TOTAL
    QUAL_TRAI bitfield;                 /* /TRAILING
    QUAL_VERS bitfield;                 /* /VERSIONS
    QUAL_WIDT bitfield;                 /* /WIDTH
    WIDT_DISP bitfield;                 /* /WIDTH=DISPLAY:n
    WIDT_FILE bitfield;                 /* /WIDTH=FILENAME:n
    WIDT_OWNE bitfield;                 /* /WIDTH=OWNER:n

    COMM_QUAL bitfield;                 /* One of the common qualifiers seen
    COLU_DEF bitfield;                  /* Column count defaulted
    NEED_FHC bitfield;                  /* Need the FHC XAB
    NEED_DAT bitfield;                  /* Need the DAT XAB
    NEED_PRO bitfield;                  /* Need the PRO XAB
    NEED_SUM bitfield;                  /* Need the SUM XAB
    NEED_JNL bitfield;                  /* Need the JNL XAB
    FILE_FOUND bitfield;                /* Files found to display
    USE_ID bitfield;                    /* Use identifiers
end QUAL_BITS;
end QUALDEF;

end_module $DIRQUALDEF;
```

DIFMSG
LIS

DIRECTORY
LIS

MAIN
LIS

DIR

DIRECTORY
MAP

DIRECTDEF
REQ

OUTPUT
LIS

DISPLYDEF
SDL

DIRECTMSG
LIS