```
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        RRRRRRRRRRRR
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
DDD          DDD       III        RRR      RRR
DDDDDDDDDDDD        IIIIIIIII     RRR         RRR
DDDDDDDDDDDD        IIIIIIIII     RRR         RRR
DDDDDDDDDDDD        IIIIIIIII     RRR         RRR
```

```
DDDDDDDD      IIIIII    RRRRRRRR    EEEEEEEEEE    CCCCCCCC  TTTTTTTTTT  DDDDDDDD    EEEEEEEEEE  FFFFFFFFFF
DDDDDDDD      IIIIII    RRRRRRRR    EEEEEEEEEE    CCCCCCCC  TTTTTTTTTT  DDDDDDDD    EEEEEEEEEE  FFFFFFFFFF
DD      DD      II      RR      RR  EE          CC              TT      DD      DD  EE          FF
DD      DD      II      RR      RR  EE          CC              TT      DD      DD  EE          FF
DD      DD      II      RR      RR  EE          CC              TT      DD      DD  EE          FF
DD      DD      II      RR      RR  EE          CC              TT      DD      DD  EE          FF
DD      DD      II      RRRRRRRR    EEEEEEEE    CC              TT      DD      DD  EEEEEEEE    FFFFFFFF
DD      DD      II      RRRRRRRR    EEEEEEEE    CC              TT      DD      DD  EEEEEEEE    FFFFFFFF
DD      DD      II      RR  RR      EE          CC              TT      DD      DD  EE          FF
DD      DD      II      RR  RR      EE          CC              TT      DD      DD  EE          FF
DD      DD      II      RR    RR    EE          CC              TT      DD      DD  EE          FF
DD      DD      II      RR    RR    EE          CC              TT      DD      DD  EE          FF
DDDDDDDD      IIIIII    RR      RR  EEEEEEEEEE    CCCCCCCC      TT      DDDDDDDD    EEEEEEEEEE  FF
DDDDDDDD      IIIIII    RR      RR  EEEEEEEEEE    CCCCCCCC      TT      DDDDDDDD    EEEEEEEEEE  FF
```

```
RRRRRRRR    EEEEEEEEEE    QQQQQQ
RRRRRRRR    EEEEEEEEEE    QQQQQQ
RR      RR  EE          QQ      QQ
RR      RR  EE          QQ      QQ
RR      RR  EE          QQ      QQ
RR      RR  EE          QQ      QQ
RRRRRRRR    EEEEEEEE    QQ      QQ
RRRRRRRR    EEEEEEEE    QQ      QQ
RR  RR      EE          QQ  QQ  QQ
RR  RR      EE          QQ  QQ  QQ
RR    RR    EE          QQ    QQ
RR    RR    EE          QQ    QQ
RR      RR  EEEEEEEEEE    QQQQ  QQ
RR      RR  EEEEEEEEEE    QQQQ  QQ
```

```
REQUIRE FILE FOR THE DIRECTORY COMMAND

Version:        'V04-000'
```



```
++

FACILITY:     DIRECTORY

ABSTRACT:

      This file defines all of the flags and storage offsets local
      to the DIRECTORY command.

ENVIRONMENT:

      VAX/VMS operating system, unprivileged user mode utilities.

--

AUTHOR:       L. Mark Pilant          CREATION DATE:  3-Mar-1983

MODIFIED BY:

      V03-008 LMP0211         L. Mark Pilant,         10-Mar-1984  13:23
              Add storage for the length of the file's ACL and the name of
              the device on which the file lives.

      V03-007 TMK0001         Todd M. Katz            13-Nov-1983
              Change all references of FH2$M_RUACCESS to FH2$M_ONLY_RU.

      V03-006 LMP0157         L. Mark Pilant,         27-Sep-1983  14:07
```

```
!               Add support for a unique message file.
!
!       V03-005 DAS0002         David Solomon           01-Aug-1983
!               Fix broken $ASSUME due to max journal name going from 12 to 16.
!
!       V03-004 DAS0001         David Solomon           29-Jul-1983
!               XAB$M_RUA is now XAB$M_ONLY_RU.
!
!       V03-003 LMP0108         L. Mark Pilant,         28-Apr-1983  11:10
!               Flush the output buffers when an error occurs.  Also add
!               support for RMS journal names.
!
!       V03-002 LMP0100         L. Mark Pilant,         14-Apr-1983  11:43
!               Misc fixups.
!
!       V03-001 LMP0096         L. Mark Pilant,         29-Mar-1983  9:58
!               Correctly handle locked files.
!
!--

REQUIRE 'LIB$:DISPLYDEF';
```

```
! Assumptions made about various constants.

$ASSUME (XAB$C_MAXJNLNAM EQL 16);
$ASSUME (XAB$M_ONLY_RU EQL FH2$M_ONLY_RU);
$ASSUME (XAB$M_RU EQL FH2$M_RUJNL);
$ASSUME (XAB$M_BI EQL FH2$M_BIJNL);
$ASSUME (XAB$M_AI EQL FH2$M_AIJNL);
$ASSUME (XAB$M_AT EQL FH2$M_ATJNL);

! Built in instructions.

BUILTIN
        ROT;

! Macro to signal an error and save the worst case error condition.

MACRO
        SIGNAL  (ERROR) =
                BEGIN
                EXTERNAL ROUTINE
                        LIB$SIGNAL : ADDRESSING_MODE (GENERAL);
                $FLUSH (RAB = OUTPUT_RAB);
                $WAIT (RAB = OUTPUT_RAB);
                LIB$SIGNAL (ERROR %IF %LENGTH-1 GTR 0 %THEN, %REMAINING %FI);
                IF (ERROR AND STS$M_SEVERITY) NEQ STS$K_WARNING
                AND (.WORST_ERROR AND STS$M_SEVERITY) LSS
                    (ERROR AND STS$M_SEVERITY) THEN WORST_ERROR = ERROR OR
                                                                 STS$M_INHIB_MSG;
                END
                %;

! Macro to do a formatted write to the selected output device/file.

MACRO
        WRITE (CODE, STRING) =
            DIR$OUTPUT (CODE
                        %IF %LENGTH GTR 1 %THEN, $DESCRIPTOR (STRING) %FI
                        %IF %LENGTH GTR 2 %THEN, %REMAINING %FI) %;

! Macro to format and append text to the current line.

MACRO
        APPEND (CODE, STRING) =
            DIR$APPEND (CODE
                        %IF %LENGTH GTR 1 %THEN, $DESCRIPTOR (STRING) %FI
                        %IF %LENGTH GTR 2 %THEN, %REMAINING %FI) %;

! Shared message definitions.

$SHR_MSGDEF     (DIR, 121, LOCAL,
                (SYNTAX, SEVERE),
                (OPENIN, ERROR),
                (OPENOUT, SEVERE),
                (WRITEERR, SEVERE),
                (CLOSEOUT, ERROR)
                );
```

```
' Common literals.

LITERAL
        DIR_C_MAXMSGSIZ = 256;                  ! Maximum message size

! Define common storage.

PSECT   OWN=DIR$COMMON (OVERLAY, ADDRESSING_MODE (LONG_RELATIVE), ALIGN (0));

OWN
        QUAL_FLAGS      : $BBLOCK [8],  ! Command qualifier flags
        COLUMN_COUNT,                   ! Number of columns in the display
        COLUMN_INDEX,                   ! Current column number
        COLUMN_WIDTH,                   ! Width of a column
        WORST_ERROR,                    ! The worst error encountered
        CMN_QUAL_CTX,                   ! Common qualifier parsing context
        DISPLAY_BLOCK   : REF $BBLOCK,  ! Address of the display block
        CHANNEL,                        ! I/O channel for information gathering
        DEVICE_NAME     : VECTOR [NAM$C_DVI, BYTE],     ! Current device name descr
        LINE_DESC       : $BBLOCK [DSC$C_S_BLN],        ! Output line descriptor
        LINE_BUFFER     : VECTOR [1024, BYTE],  ! Output line text storage
        TOTAL_USED,                     ! Total block used
        TOTAL_ALLOC,                    ! Total blocks allocated
        TOTAL_FILES,                    ! Total files in directory
        GRAND_USED,                     ! Grand total blocks used
        GRAND_ALLOC,                    ! Grand total block allocated
        GRAND_FILES,                    ! Total files listed
        GRAND_DIRS,                     ! Total directories listed
        PREV_DIR        : VECTOR [NAM$C_MAXRSS, BYTE],  ! Previous dir name
        PREV_DIR_LEN,                   ! Previous directory name length
        PREV_FILE       : VECTOR [NAM$C_MAXRSS, BYTE],  ! Previous file name
        PREV_FILE_LEN,                  ! Previous file name length
        VERSION_COUNT,                  ! Number of versions to list
        VERSION_INDEX,                  ! Current version
        FIRST_XAB       : REF $BBLOCK,  ! Address of the first XAB
        INFO_XABJNL     : $XABJNL (),   ! RMS journaling information
        INFO_XABSUM     : $XABSUM (),   ! RMS summary information
        INFO_XABPRO     : $XABPRO (),   ! File protection info
        INFO_XABDAT     : $XABDAT (),   ! File dates
        INFO_XABFHC     : $XABFHC (),   ! File header data
        INFO_NAM        : $NAM (),      ! NAMe block for opening file
        INFO_FAB        : $FAB (FOP = NAM,      ! FAB for getting file info
                                NAM = INFO_NAM,
                                SHR = <GET, PUT, UPI>),
        DISPLAY_WIDTH,                  ! Max width of the display
        FILENAME_WIDTH,                 ! Max width of the file name
        OWNER_WIDTH,                    ! Max width of the owner (UIC)
        SIZE_WIDTH,                     ! Max width of the size field (blocks)
        MIN_BLOCK,                      ! Low end block size selection limit
        MAX_BLOCK,                      ! High end block size selection limit
        ACL_LENGTH,                     ! Length of the file's ACL
        OUTPUT_RAB      : $RAB_DECL;    ! Output file RAB

PSECT   OWN=$OWN$ (CONCATENATE, ADDRESSING_MODE (WORD_RELATIVE), ALIGN (2));
```

DIFMSG
LIS

DIRECTORY
LIS

MAIN
LIS

DIR

DIRECTORY
MAP

DIRECTDEF
REQ

OUTPUT
LIS

DISPLYDEF
SDL

DIRECTMSG
LIS