```
;
;       Z-8Ø MACRO LIBRARY
;
;    THE FOLLOWING MACROS ENABLE ASSEMBLING Z-8Ø INSTRUCTIONS
;    WITH THE DIGITAL RESEARCH MACRO ASSEMBLER.
;
;  INVOKE WITH "MACLIB Z8Ø"
;
;
;
;                       MACRO FORMATS
;                       ----- -------
;
;
;       MACRO                   ZILOG                   TDL
;       -----                   -----                   ---
;
;       LDX     R,D             LD      R,(IX+D)        MOV     R,D(IX)
;       LDY     R,D             LD      R,(IY+D)        MOV     R,D(IY)
;       STX     R,D             LD      (IX+D),R        MOV     D(IX),R
;       STY     R,D             LD      (IY+D),R        MOV     D(IY),R
;       MVIX    NN,D            LD      (IX+D),NN       MVI     D(IX)
;       MVIY    NN,D            LD      (IY+D),NN       MVI     D(IY)
;       LDAI                    LD      A,I             LDAI
;       LDAR                    LD      A,R             LDAR
;       STAI                    LD      I,A             STAI
;       STAR                    LD      R,A             STAR
;       LXIX    NNNN            LD      IX,NNNN         LXI     IX,NNNN
;       LXIY    NNNN            LD      IY,NNNN         LXI     IY,NNNN
;       LBCD    NNNN            LD      BC,(NNNN)       LBCD    NNNN
;       LDED    NNNN            LD      DE,(NNNN)       LDED    NNNN
;       LSPD    NNNN            LD      SP,(NNNN)       LSPD    NNNN
;       LIXD    NNNN            LD      IX,(NNNN)       LIXD    NNNN
;       LIYD    NNNN            LD      IY,(NNNN)       LIYD    NNNN
;       SBCD    NNNN            LD      (NNNN),BC       SBCD    NNNN
;       SDED    NNNN            LD      (NNNN),DE       SDED    NNNN
;       SSPD    NNNN            LD      (NNNN),SP       SSPD    NNNN
;       SIXD    NNNN            LD      (NNNN),IX       SIXD    NNNN
;       SIYD    NNNN            LD      (NNNN),IY       SIYD    NNNN
;       SPIX                    LD      SP,IX           SPIX
;       SPIY                    LD      SP,IY           SPIY
;       PUSHIX                  PUSH    IX              PUSH    IX
;       PUSHIY                  PUSH    IY              PUSH    IY
;       POPIX                   POP     IX              POP     IX
;       POPIY                   POP     IY              POP     IY
;       EXAF                    EX      AF,AF'          EXAF
;       EXX                     EXX                     EXX
;       XTIX                    EX      (SP),IX         XTIX
;       XTIY                    EX      (SP),IY         XTIY
;       LDI                     LDI                     LDI
;       LDIR                    LDIR                    LDIR
;       LDD                     LDD                     LDD
;       LDDR                    LDDR                    LDDR
;       CCI                     CPI                     CCI
;       CCIR                    CPIR                    CCIR
;       CCD                     CPD                     CCD
;       CCDR                    CPDR                    CCDR
;       ADDX    D               ADD     (IX+D)          ADD     D(IX)
;       ADDY    D               ADD     (IY+D)          ADD     D(IY)
;       ADCX    D               ADC     (IX+D)          ADC     D(IX)
```

```
      ;         ADCY      D                   ADC       (IY+D)              ADC       D(IY)
      ;         SUBX      D                   SUB       (IX+D)              SUB       D(IX)
      ;         SUBY      D                   SUB       (IY+D)              SUB       D(IY)
      ;         SBCX      D                   SBC       (IX+D)              SBB       D(IX)
      ;         SBCY      D                   SBC       (IY+D)              SBB       D(IY)
      ;         ANDX      D                   AND       (IX+D)              ANA       D(IX)
      ;         ANDY      D                   AND       (IY+D)              ANA       D(IY)
      ;         XORX      D                   XOR       (IX+D)              XRA       D(IX)
      ;         XORY      D                   XOR       (IY+D)              XRA       D(IY)
      ;         ORX       D                   OR        (IX+D)              ORA       D(IX)
      ;         ORY       D                   OR        (IY+D)              ORA       D(IY)
      ;         CMPX      D                   CP        (IX+D)              CMP       D(IX)
      ;         CMPY      D                   CP        (IY+D)              CMP       D(IY)
      ;         INRX      D                   INC       (IX+D)              INR       D(IX)
      ;         INRY      D                   INC       (IY+D)              INR       D(IY)
      ;         DCRX      D                   INC       (IX+D)              INR       D(IX)
      ;         DCRY      D                   DEC       (IY+D)              DCR       D(IY)
      ;         NEG                           NEG                           NEG
      ;         IM0                           IM0                           IM0
      ;         IM1                           IM1                           IM1
      ;         IM2                           IM2                           IM2
      ;         DADC      RR                  ADC       HL,RR               DADC      RR
      ;         DSBC      RR                  SBC       HL,RR               DSBC      RR
      ;         DADX      RR                  ADD       IX,RR               DADX      RR
      ;         DADY      RR                  ADD       IY,RR               DADY      RR
      ;         INXIX                         INC       IX                  INX       IX
      ;         INXIY                         INC       IY                  INX       IY
      ;         DCXIX                         DEC       IX                  DCX       IX
      ;         DCXIY                         DEC       IY                  DCX       IY
      ;         BIT       B,R                 BIT       B,R                 BIT       B,R
      ;         SETB      B,R                 SET       B,R                 SET       B,R
      ;         RES       B,R                 RES       B,R                 RES       B,R
      ;         BITX      B,D                 BIT       B,(IX+D)            BIT       B,D(IX)
      ;         BITY      B,D                 BIT       B,(IY+D)            BIT       B,D(IY)
      ;         SETX      B,D                 SET       B,(IX+D)            SET       B,D(IX)
      ;         SETY      B,D                 SET       B,(IY+D)            SET       B,D(IY)
      ;         RESX      B,D                 RES       B,(IX+D)            RES       B,D(IX)
      ;         RESY      B,D                 RES       B,(IY+D)            RES       B,D(IY)
      ;         JR        ADDR                JR        ADDR-$              JMPR      ADDR
      ;         JRC       ADDR                JR        C,ADDR-$            JRC       ADDR
      ;         JRNC      ADDR                JR        NC,ADDR-$           JRNC      ADDR
      ;         JRZ       ADDR                JR        Z,ADDR-$            JRC       ADDR
      ;         JRNZ      ADDR                JR        NZ,ADDR-$           JRNZ      ADDR
      ;         DJNZ      ADDR                DJNZ      ADDR-$              DJNZ      ADDR
      ;         PCIX                          JMP       (IX)                PCIX
      ;         PCIY                          JMP       (IY)                PCIY
      ;         RETI                          RETI                          RETI
      ;         RETN                          RETN                          RETN
      ;         INP       R                   IN        R,(C)               INP       R
      ;         OUTP      R                   OUT       (C),R               OUTP      R
      ;         INI                           INI                           INI
      ;         INIR                          INIR                          INIR
      ;         OUTI                          OTI                           OUTI
      ;         OUTIR                         OTIR                          OUTIR
      ;         IND                           IND                           IND
      ;         INDR                          INDR                          INDR
      ;         OUTD                          OTD                           OUTD
      ;         OUTDR                         OTDR                          OUTDR
      ;         RLCR      R                   RLC       R                   RLCR      R
      ;         RLCX      D                   RLC       (IX+D)              RLCR      D(IX)
      ;         RLCY      D                   RLC       (IY+D)              RLCR      D(IY)
      ;         RALR      R                   RL        R                   RALR      R
      ;         RALX      D                   RL        (IX+D)              RALR      D(IX)
      ;         RALY      D                   RL        (IY+D)              RALR      D(IY)
      ;         RRCR      R                   RRC       R                   RRCR      R
      ;         RRCX      D                   RRC       (IX+D)              RRCR      D(IX)
```

```
;       RRCY    D               RRC     (IY+D)          RRCR    D(IY)
;       RARR    R               RR      R               RARR    R
;       RARX    D               RR      (IX+D)          RARR    D(IX)
;       RARY    D               RR      (IY+D)          RARR    D(IY)
;       SLAR    R               SLA     R               SLAR    R
;       SLAX    D               SLA     (IX+D)          SLAR    D(IX)
;       SLAY    D               SLA     (IY+D)          SLAR    D(IY)
;       SRAR    R               SRA     R               SRAR    R
;       SRAX    D               SRA     (IX+D)          SRAR    D(IX)
;       SRAY    D               SRA     (IY+D)          SRAR    D(IY)
;       SRLR    R               SRL     R               SRLR    R
;       SRLX    D               SRL     (IX+D)          SRLR    D(IX)
;       SRLY    D               SRL     (IY+D)          SRLR    D(IY)
;       RLD                     RLD                     RLD
;       RRD                     RRD                     RRD
;
;
;
LDX     MACRO   ?R,?D
        DB      ØDDH,?R*8+46H,?D
        ENDM
LDY     MACRO   ?R,?D
        DB      ØFDH,?R*8+46H,?D
        ENDM
STX     MACRO   ?R,?D
        DB      ØDDH,7ØH+?R,?D
        ENDM
STY     MACRO   ?R,?D
        DB      ØFDH,7ØH+?R,?D
        ENDM
MVIX    MACRO   ?N,?D
        DB      ØDDH,36H,?D,?N
        ENDM
MVIY    MACRO   ?N,?D
        DB      ØFDH,36H,?D,?N
        ENDM
LDAI    MACRO
        DB      ØEDH,57H
        ENDM
LDAR    MACRO
        DB      ØEDH,5FH
        ENDM
STAI    MACRO
        DB      ØEDH,47H
        ENDM
STAR    MACRO
        DB      ØEDH,4FH
        ENDM

LXIX    MACRO   ?NNNN
        DB      ØDDH,21H
        DW      ?NNNN
        ENDM
LXIY    MACRO   ?NNNN
        DB      ØFDH,21H
        DW      ?NNNN
        ENDM
LDED    MACRO   ?NNNN
        DB      ØEDH,5BH
        DW      ?NNNN
        ENDM
LBCD    MACRO   ?NNNN
        DB      ØEDH,4BH
        DW      ?NNNN
        ENDM
LSPD    MACRO   ?NNNN
```

```
        DB      ØEDH,Ø7BH
        DW      ?NNNN
        ENDM
LIXD    MACRO   ?NNNN
        DB      ØDDH,2AH
        DW      ?NNNN
        ENDM
LIYD    MACRO   ?NNNN
        DB      ØFDH,2AH
        DW      ?NNNN
        ENDM
SBCD    MACRO   ?NNNN
        DB      ØEDH,43H
        DW      ?NNNN
        ENDM
SDED    MACRO   ?NNNN
        DB      ØEDH,53H
        DW      ?NNNN
        ENDM
SSPD    MACRO   ?NNNN
        DB      ØEDH,73H
        DW      ?NNNN
        ENDM
SIXD    MACRO   ?NNNN
        DB      ØDDH,22H
        DW      ?NNNN
        ENDM
SIYD    MACRO   ?NNNN
        DB      ØFDH,22H
        DW      ?NNNN
        ENDM
SPIX    MACRO
        DB      ØDDH,ØF9H
        ENDM
SPIY    MACRO
        DB      ØFDH,ØF9H
        ENDM
PUSHIX  MACRO
        DB      ØDDH,ØE5H
        ENDM
PUSHIY  MACRO
        DB      ØFDH,ØE5H
        ENDM
POPIX   MACRO
        DB      ØDDH,ØE1H
        ENDM
POPIY   MACRO
        DB      ØFDH,ØE1H
        ENDM
EXAF    MACRO
        DB      Ø8H
        ENDM
EXX     MACRO
        DB      ØD9H
        ENDM
XTIX    MACRO
        DB      ØDDH,ØE3H
        ENDM
XTIY    MACRO
        DB      ØFDH,ØE3H
        ENDM

LDI     MACRO
        DB      ØEDH,ØAØH
        ENDM
LDIR    MACRO
```

```
        DB      ØEDH,ØBØH
        ENDM
LDD     MACRO
        DB      ØEDH,ØA8H
        ENDM
LDDR    MACRO
        DB      ØEDH,ØB8H
        ENDM
CCI     MACRO
        DB      ØEDH,ØA1H
        ENDM
CCIR    MACRO
        DB      ØEDH,ØB1H
        ENDM
CCD     MACRO
        DB      ØEDH,ØA9H
        ENDM
CCDR    MACRO
        DB      ØEDH,ØB9H
        ENDM

ADDX    MACRO   ?D
        DB      ØDDH,86H,?D
        ENDM
ADDY    MACRO   ?D
        DB      ØFDH,86H,?D
        ENDM
ADCX    MACRO   ?D
        DB      ØDDH,8EH,?D
        ENDM
ADCY    MACRO   ?D
        DB      ØFDH,8EH,?D
        ENDM
SUBX    MACRO   ?D
        DB      ØDDH,96H,?D
        ENDM
SUBY    MACRO   ?D
        DB      ØFDH,96H,?D
        ENDM
SBCX    MACRO   ?D
        DB      ØDDH,9EH,?D
        ENDM
SBCY    MACRO   ?D
        DB      ØFDH,9EH,?D
        ENDM
ANDX    MACRO   ?D
        DB      ØDDH,ØA6H,?D
        ENDM
ANDY    MACRO   ?D
        DB      ØFDH,ØA6H,?D
        ENDM
XORX    MACRO   ?D
        DB      ØDDH,ØAEH,?D
        ENDM
XORY    MACRO   ?D
        DB      ØFDH,ØAEH,?D
        ENDM
ORX     MACRO   ?D
        DB      ØDDH,ØB6H,?D
        ENDM
ORY     MACRO   ?D
        DB      ØFDH,ØB6H,?D
        ENDM
CMPX    MACRO   ?D
        DB      ØDDH,ØBEH,?D
        ENDM
```

```
CMPY    MACRO   ?D
        DB      ØFDH,ØBEH,?D
        ENDM
INRX    MACRO   ?D
        DB      ØDDH,34H,?D
        ENDM
INRY    MACRO   ?D
        DB      ØFDH,34H,?D
        ENDM
DCRX    MACRO   ?D
        DB      ØDDH,Ø35H,?D
        ENDM
DCRY    MACRO   ?D
        DB      ØFDH,35H,?D
        ENDM

NEG     MACRO
        DB      ØEDH,44H
        ENDM
IMØ     MACRO
        DB      ØEDH,46H
        ENDM
IM1     MACRO
        DB      ØEDH,56H
        ENDM
IM2     MACRO
        DB      ØEDH,5EH
        ENDM


BC      EQU     Ø
DE      EQU     2
HL      EQU     4
IX      EQU     4
IY      EQU     4
DADC    MACRO   ?R
        DB      ØEDH,?R*8+4AH
        ENDM
DSBC    MACRO   ?R
        DB      ØEDH,?R*8+42H
        ENDM
DADX    MACRO   ?R
        DB      ØDDH,?R*8+Ø9H
        ENDM
DADY    MACRO   ?R
        DB      ØFDH,?R*8+Ø9H
        ENDM
INXIX   MACRO
        DB      ØDDH,23H
        ENDM
INXIY   MACRO
        DB      ØFDH,23H
        ENDM
DCXIX   MACRO
        DB      ØDDH,2BH
        ENDM
DCXIY   MACRO
        DB      ØFDH,2BH
        ENDM

BIT     MACRO   ?N,?R
        DB      ØCBH,?N*8+?R+4ØH
        ENDM
SETB    MACRO   ?N,?R
        DB      ØCBH,?N*8+?R+ØCØH
        ENDM
```

```
RES     MACRO   ?N,?R
        DB      ØCBH,?N*8+?R+8ØH
        ENDM

BITX    MACRO   ?N,?D
        DB      ØDDH,ØCBH,?D,?N*8+46H
        ENDM
BITY    MACRO   ?N,?D
        DB      ØFDH,ØCBH,?D,?N*8+46H
        ENDM
SETX    MACRO   ?N,?D
        DB      ØDDH,ØCBH,?D,?N*8+ØC6H
        ENDM
SETY    MACRO   ?N,?D
        DB      ØFDH,ØCBH,?D,?N*8+ØC6H
        ENDM
RESX    MACRO   ?N,?D
        DB      ØDDH,ØCBH,?D,?N*8+86H
        ENDM
RESY    MACRO   ?N,?D
        DB      ØFDH,ØCBH,?D,?N*8+86H
        ENDM

JR      MACRO   ?N
        DB      18H,?N-$-1
        ENDM
JRC     MACRO   ?N
        DB      38H,?N-$-1
        ENDM
JRNC    MACRO   ?N
        DB      3ØH,?N-$-1
        ENDM
JRZ     MACRO   ?N
        DB      28H,?N-$-1
        ENDM
JRNZ    MACRO   ?N
        DB      2ØH,?N-$-1
        ENDM
DJNZ    MACRO   ?N
        DB      1ØH,?N-$-1
        ENDM

PCIX    MACRO
        DB      ØDDH,ØE9H
        ENDM
PCIY    MACRO
        DB      ØFDH,ØE9H
        ENDM

RETI    MACRO
        DB      ØEDH,4DH
        ENDM
RETN    MACRO
        DB      ØEDH,45H
        ENDM

INP     MACRO   ?R
        DB      ØEDH,?R*8+4ØH
        ENDM
OUTP    MACRO   ?R
        DB      ØEDH,?R*8+41H
        ENDM
INI     MACRO
        DB      ØEDH,ØA2H
        ENDM
INIR    MACRO
```

```
          DB        ØEDH,ØB2H
          ENDM
IND       MACRO
          DB        ØEDH,ØAAH
          ENDM
INDR      MACRO
          DB        ØEDH,ØBAH
          ENDM
OUTI      MACRO
          DB        ØEDH,ØA3H
          ENDM
OUTIR     MACRO
          DB        ØEDH,ØB3H
          ENDM
OUTD      MACRO
          DB        ØEDH,ØABH
          ENDM
OUTDR     MACRO
          DB        ØEDH,ØBBH
          ENDM


RLCR      MACRO     ?R
          DB        ØCBH, ØØH + ?R
          ENDM
RLCX      MACRO     ?D
          DB        ØDDH, ØCBH, ?D, Ø6H
          ENDM
RLCY      MACRO     ?D
          DB        ØFDH, ØCBH, ?D, Ø6H
          ENDM
RALR      MACRO     ?R
          DB        ØCBH, 1ØH+?R
          ENDM
RALX      MACRO     ?D
          DB        ØDDH, ØCBH, ?D, 16H
          ENDM
RALY      MACRO     ?D
          DB        ØFDH, ØCBH, ?D, 16H
          ENDM
RRCR      MACRO     ?R
          DB        ØCBH, Ø8H + ?R
          ENDM
RRCX      MACRO     ?D
          DB        ØDDH, ØCBH, ?D, ØEH
          ENDM
RRCY      MACRO     ?D
          DB        ØFDH, ØCBH, ?D, ØEH
          ENDM
RARR      MACRO     ?R
          DB        ØCBH, 18H + ?R
          ENDM
RARX      MACRO     ?D
          DB        ØDDH, ØCBH, ?D, 1EH
          ENDM
RARY      MACRO     ?D
          DB        ØFDH, ØCBH, ?D, 1EH
          ENDM
SLAR      MACRO     ?R
          DB        ØCBH, 2ØH + ?R
          ENDM
SLAX      MACRO     ?D
          DB        ØDDH, ØCBH, ?D, 26H
          ENDM
SLAY      MACRO     ?D
          DB        ØFDH, ØCBH, ?D, 26H
```

```
        ENDM
SRAR    MACRO   ?R
        DB      ØCBH, 28H+?R
        ENDM
SRAX    MACRO   ?D
        DB      ØDDH, ØCBH, ?D, 2EH
        ENDM
SRAY    MACRO   ?D
        DB      ØFDH, ØCBH, ?D, 2EH
        ENDM
SRLR    MACRO   ?R
        DB      ØCBH, 38H + ?R
        ENDM
SRLX    MACRO   ?D
        DB      ØDDH, ØCBH, ?D, 3EH
        ENDM
SRLY    MACRO   ?D
        DB      ØFDH, ØCBH, ?D, 3EH
        ENDM
RLD     MACRO
        DB      ØEDH, 6FH
        ENDM
RRD     MACRO
        DB      ØEDH, 67H
        ENDM
;       CP/M 2.Ø DISK RE-DEFINITION LIBRARY
;
;       COPYRIGHT (C) 1979
;       DIGITAL RESEARCH
;       BOX 579
;       PACIFIC GROVE, CA
;       9395Ø
;
;       CP/M LOGICAL DISK DRIVES ARE DEFINED USING THE
;       MACROS GIVEN BELOW, WHERE THE SEQUENCE OF CALLS
;       IS:
;
;       DISKS   N
;       DISKDEF PARAMETER-LIST-Ø
;       DISKDEF PARAMETER-LIST-1
;       ...
;       DISKDEF PARAMETER-LIST-N
;       ENDEF
;
;       WHERE N IS THE NUMBER OF LOGICAL DISK DRIVES ATTACHED
;       TO THE CP/M SYSTEM, AND PARAMETER-LIST-I DEFINES THE
;       CHARACTERISTICS OF THE ITH DRIVE (I=Ø,1,...,N-1)
;
;       EACH PARAMETER-LIST-I TAKES THE FORM
;               DN,FSC,LSC,[SKF],BLS,DKS,DIR,CKS,OFS,[Ø]
;       WHERE
;       DN      IS THE DISK NUMBER Ø,1,...,N-1
;       FSC     IS THE FIRST SECTOR NUMBER (USUALLY Ø OR 1)
;       LSC     IS THE LAST SECTOR NUMBER ON A TRACK
;       SKF     IS OPTIONAL "SKEW FACTOR" FOR SECTOR TRANSLATE
;       BLS     IS THE DATA BLOCK SIZE (1Ø24,2Ø48,...,16384)
;       DKS     IS THE DISK SIZE IN BLS INCREMENTS (WORD)
;       DIR     IS THE NUMBER OF DIRECTORY ELEMENTS (WORD)
;       CKS     IS THE NUMBER OF DIR ELEMENTS TO CHECKSUM
;       OFS     IS THE NUMBER OF TRACKS TO SKIP (WORD)
;       [Ø]     IS AN OPTIONAL Ø WHICH FORCES 16K/DIRECTORY ENTRY
;
;       FOR CONVENIENCE, THE FORM
;               DN,DM
;       DEFINES DISK DN AS HAVING THE SAME CHARACTERISTICS AS
;       A PREVIOUSLY DEFINED DISK DM.
```

```
;
;       A STANDARD FOUR DRIVE CP/M SYSTEM IS DEFINED BY
;               DISKS   4
;               DISKDEF 0,1,26,6,1024,243,64,64,2
;       DSK     SET     0
;               REPT    3
;       DSK     SET     DSK+1
;               DISKDEF %DSK,0
;               ENDM
;               ENDEF
;
;       THE VALUE OF "BEGDAT" AT THE END OF ASSEMBLY DEFINES THE
;       BEGINNING OF THE UNINITIALIZE RAM AREA ABOVE THE BIOS,
;       WHILE THE VALUE OF "ENDDAT" DEFINES THE NEXT LOCATION
;       FOLLOWING THE END OF THE DATA AREA.  THE SIZE OF THIS
;       AREA IS GIVEN BY THE VALUE OF "DATSIZ" AT THE END OF THE
;       ASSEMBLY.  NOTE THAT THE ALLOCATION VECTOR WILL BE QUITE
;       LARGE IF A LARGE DISK SIZE IS DEFINED WITH A SMALL BLOCK
;       SIZE.
;
DSKHDR  MACRO   DN
;;      DEFINE A SINGLE DISK HEADER LIST
DPE&DN: DW      XLT&DN,0000H    ;TRANSLATE TABLE
        DW      0000H,0000H     ;SCRATCH AREA
        DW      DIRBUF,DPB&DN   ;DIR BUFF,PARM BLOCK
        DW      CSV&DN,ALV&DN   ;CHECK, ALLOC VECTORS
        ENDM
;
DISKS   MACRO   ND
;;      DEFINE ND DISKS
NDISKS  SET     ND      ;;FOR LATER REFERENCE
DPBASE  EQU     $        ;BASE OF DISK PARAMETER BLOCKS
;;      GENERATE THE ND ELEMENTS
DSKNXT  SET     0
        REPT    ND
        DSKHDR  %DSKNXT
DSKNXT  SET     DSKNXT+1
        ENDM
        ENDM
;
DPBHDR  MACRO   DN
DPB&DN  EQU     $               ;DISK PARM BLOCK
        ENDM
;
DDB     MACRO   DATA,COMMENT
;;      DEFINE A DB STATEMENT
        DB      DATA            COMMENT
        ENDM
;
DDW     MACRO   DATA,COMMENT
;;      DEFINE A DW STATEMENT
        DW      DATA            COMMENT
        ENDM
;
GCD     MACRO   M,N
;;      GREATEST COMMON DIVISOR OF M,N
;;      PRODUCES VALUE GCDN AS RESULT
;;      (USED IN SECTOR TRANSLATE TABLE GENERATION)
GCDM    SET     M       ;;VARIABLE FOR M
GCDN    SET     N       ;;VARIABLE FOR N
GCDR    SET     0       ;;VARIABLE FOR R
        REPT    65535
GCDX    SET     GCDM/GCDN
GCDR    SET     GCDM - GCDX*GCDN
        IF      GCDR = 0
        EXITM
        ENDIF
GCDM    SET     GCDN
GCDN    SET     GCDR
        ENDM
        ENDM
;
DISKDEF MACRO   DN,FSC,LSC,SKF,BLS,DKS,DIR,CKS,OFS,K16
;;      GENERATE THE SET STATEMENTS FOR LATER TABLES
        IF      NUL LSC
;;      CURRENT DISK DN SAME AS PREVIOUS FSC
DPB&DN  EQU     DPB&FSC ;EQUIVALENT PARAMETERS
ALS&DN  EQU     ALS&FSC ;SAME ALLOCATION VECTOR SIZE
CSS&DN  EQU     CSS&FSC ;SAME CHECKSUM VECTOR SIZE
XLT&DN  EQU     XLT&FSC ;SAME TRANSLATE TABLE
        ELSE
SECMAX  SET     LSC-(FSC)       ;;SECTORS Ø...SECMAX
SECTORS SET     SECMAX+1;;NUMBER OF SECTORS
ALS&DN  SET     (DKS)/8 ;;SIZE OF ALLOCATION VECTOR
        IF      ((DKS) MOD 8) NE Ø
ALS&DN  SET     ALS&DN+1
        ENDIF
CSS&DN  SET     (CKS)/4 ;;NUMBER OF CHECKSUM ELEMENTS
;;      GENERATE THE BLOCK SHIFT VALUE
BLKVAL  SET     BLS/128 ;;NUMBER OF SECTORS/BLOCK
BLKSHF  SET     Ø       ;;COUNTS RIGHT Ø'S IN BLKVAL
BLKMSK  SET     Ø       ;;FILLS WITH 1'S FROM RIGHT
        REPT    16      ;;ONCE FOR EACH BIT POSITION
        IF      BLKVAL=1
        EXITM
        ENDIF
;;      OTHERWISE, HIGH ORDER 1 NOT FOUND YET
BLKSHF  SET     BLKSHF+1
BLKMSK  SET     (BLKMSK SHL 1) OR 1
BLKVAL  SET     BLKVAL/2
        ENDM
;;      GENERATE THE EXTENT MASK BYTE
BLKVAL  SET     BLS/1024        ;;NUMBER OF KILOBYTES/BLOCK
EXTMSK  SET     Ø       ;;FILL FROM RIGHT WITH 1'S
        REPT    16
        IF      BLKVAL=1
        EXITM
        ENDIF
;;      OTHERWISE MORE TO SHIFT
EXTMSK  SET     (EXTMSK SHL 1) OR 1
BLKVAL  SET     BLKVAL/2
        ENDM
;;      MAY BE DOUBLE BYTE ALLOCATION
        IF      (DKS) > 256
EXTMSK  SET     (EXTMSK SHR 1)
        ENDIF
;;      MAY BE OPTIONAL [Ø] IN LAST POSITION
        IF      NOT NUL K16
EXTMSK  SET     K16
        ENDIF
;;      NOW GENERATE DIRECTORY RESERVATION BIT VECTOR
DIRREM  SET     DIR     ;;# REMAINING TO PROCESS
DIRBKS  SET     BLS/32  ;;NUMBER OF ENTRIES PER BLOCK
DIRBLK  SET     Ø       ;;FILL WITH 1'S ON EACH LOOP
        REPT    16
        IF      DIRREM=Ø
        EXITM
        ENDIF
;;      NOT COMPLETE, ITERATE ONCE AGAIN
;;      SHIFT RIGHT AND ADD 1 HIGH ORDER BIT
DIRBLK  SET     (DIRBLK SHR 1) OR 8ØØØH
        IF      DIRREM > DIRBKS
```

```
DIRREM  SET     DIRREM-DIRBKS
        ELSE
DIRREM  SET     Ø
        ENDIF
        ENDM
        DPBHDR  DN      ;;GENERATE EQU $
        DDW     %SECTORS,<;SEC PER TRACK>
        DDB     %BLKSHF,<;BLOCK SHIFT>
        DDB     %BLKMSK,<;BLOCK MASK>
        DDB     %EXTMSK,<;EXTNT MASK>
        DDW     %(DKS)-1,<;DISK SIZE-1>
        DDW     %(DIR)-1,<;DIRECTORY MAX>
        DDB     %DIRBLK SHR 8,<;ALLOCØ>
        DDB     %DIRBLK AND ØFFH,<;ALLOC1>
        DDW     %(CKS)/4,<;CHECK SIZE>
        DDW     %OFS,<;OFFSET>
;;      GENERATE THE TRANSLATE TABLE, IF REQUESTED
        IF      NUL SKF
XLT&DN  EQU     Ø                       ;NO XLATE TABLE
        ELSE
        IF      SKF = Ø
XLT&DN  EQU     Ø                       ;NO XLATE TABLE
        ELSE
;;      GENERATE THE TRANSLATE TABLE
NXTSEC  SET     Ø       ;;NEXT SECTOR TO FILL
NXTBAS  SET     Ø       ;;MOVES BY ONE ON OVERFLOW
        GCD     %SECTORS,SKF
;;      GCDN = GCD(SECTORS,SKEW)
NELTST  SET     SECTORS/GCDN
;;      NELTST IS NUMBER OF ELEMENTS TO GENERATE
;;      BEFORE WE OVERLAP PREVIOUS ELEMENTS
NELTS   SET     NELTST  ;;COUNTER
XLT&DN  EQU     $                       ;TRANSLATE TABLE
        REPT    SECTORS ;;ONCE FOR EACH SECTOR
        IF      SECTORS < 256
        DDB     %NXTSEC+(FSC)
        ELSE
        DDW     %NXTSEC+(FSC)
        ENDIF
NXTSEC  SET     NXTSEC+(SKF)
        IF      NXTSEC >= SECTORS
NXTSEC  SET     NXTSEC-SECTORS
        ENDIF
NELTS   SET     NELTS-1
        IF      NELTS = Ø
NXTBAS  SET     NXTBAS+1
NXTSEC  SET     NXTBAS
NELTS   SET     NELTST
        ENDIF
        ENDM
        ENDIF   ;;END OF NUL FAC TEST
        ENDIF   ;;END OF NUL BLS TEST
        ENDM
;
DEFDS   MACRO   LAB,SPACE
LAB:    DS      SPACE
        ENDM
;
LDS     MACRO   LB,DN,VAL
        DEFDS   LB&DN,%VAL&DN
        ENDM
;
ENDEF   MACRO
;;      GENERATE THE NECESSARY RAM DATA AREAS
BEGDAT  EQU     $
DIRBUF: DS      128     ;DIRECTORY ACCESS BUFFER
```

```
DSKNXT  SET     0
        REPT    NDISKS  ;;ONCE FOR EACH DISK
        LDS     ALV,%DSKNXT,ALS
        LDS     CSV,%DSKNXT,CSS
DSKNXT  SET     DSKNXT+1
        ENDM
ENDDAT  EQU     $
DATSIZ  EQU     $-BEGDAT
;;      DB 0 AT THIS POINT FORCES HEX RECORD
        ENDM
;
;*******************************
;   SECTOR DEBLOCKING ALGORITHMS
;*******************************
;
;  UTILITY MACRO TO COMPUTE SECTOR MASK
;
SMASK   MACRO   HBLK
;;      COMPUTE LOG2(HBLK), RETURN @X AS RESULT
;;      (2 ** @X = HBLK ON RETURN)
@Y      SET     HBLK
@X      SET     0
;;      COUNT RIGHT SHIFTS OF @Y UNTIL = 1
        REPT    8
        IF      @Y=1
        EXITM
        ENDIF
;;      @Y IS NOT 1, SHIFT RIGHT ONE POSTION
@Y      SET     @Y SHR 1
@X      SET     @X+1
        ENDM
        ENDM
```