```
LINE# LOC   CODE        LINE

0002  0000                     .LIB COMMON
```

```
0000              ;*********************************************
0000              ;*                                           *
0000              ;* COMMODORE BUSINESS MACHINES SOFTWARE      *
0000              ;*                                           *
0000              ;*********************************************
0000              ;
0000              ;*********************************************
0000              ;*                                           *
0000              ;* DISK OPERATING SYSTEM AND                 *
0000              ;* CONTROLLER ROUTINES                       *
0000              ;* FOR THE FOLLOWING CBM MODELS:             *
0000              ;* 2031 IEEE 488 DISK                        *
0000              ;* WITH SINGLE SA390 COMPATABLE DRIVE        *
0000              ;* COPYRIGHT (C) 1982 BY                     *
0000              ;* COMMODORE BUSINESS MACHINES (CBM)         *
0000              ;*                                           *
0000              ;*********************************************
0000              ;
0000              ; ****LISTING DATE 18 JAN 1984 ****
0000              ;
0000              ;*********************************************
0000              ;* THIS SOFTWARE IS FURNISHED FOR USE IN     *
0000              ;* THE SINGLE DRIVE FLOPPY DISK UNIT ONLY.   *
0000              ;*                                           *
0000              ;* COPIES THEREOF MAY NOT BE PROVIDED OR     *
0000              ;* MADE AVAILABLE FOR USE ON ANY OTHER       *
0000              ;* SYSTEM.                                   *
0000              ;*                                           *
0000              ;* THE INFORMATION IN THIS DOCUMENT IS       *
0000              ;* SUBJECT TO CHANGE WITHOUT NOTICE.         *
0000              ;*                                           *
0000              ;* NO RESPONSIBILITY IS ASSUMED FOR          *
0000              ;* RELIABILITY OF THIS SOFTWARE. RSR         *
0000              ;*                                           *
0000              ;*********************************************
0000              ;
0000              ;*********************************************
0000              ;*                                           *
0000              ;* ASSEMBLES THE CONTENTS OF THE ROMS:       *
0000              ;* $C - $D: 901484-03                        *
0000              ;* $E - $F: 901484-05                        *
0000              ;*                                           *
0000              ;* RECONSTRUCTED BY NICO DE VRIES ON A CBM   *
0000              ;* SYSTEM CONSISTING OF A 8032 AND A 8050,   *
0000              ;* USING THE PET RESIDENT ASSEMBLER V121579  *
0000              ;* AND SURVIVING LISTINGS.                   *
0000              ;*                                           *
0000              ;*********************************************
```

```
LINE# LOC   CODE         LINE


0053  0000               ; COMMON AREA DEFINES
0054  0000               ;
0055  0000                      *=0
0056  0000               JOBS   *=*+6           ; JOB QUE
0057  0006               HDRS   *=*+12          ; JOB HEADERS
0058  0012               DSKID  *=*+4           ; MASTER COPY OF DISK ID
0059  0016               HEADER *=*+5           ; IMAGE OF LAST HEADER
0060  001B               ACTJOB *=*+1           ; CONTROLLER'S ACTIVE JOB
0061  001C               WPSW   *=*+2           ; WRITE PROTECT CHANGE FLAG
0062  001E               LWPT   *=*+2           ; LAST STATE OF WP SWITCH
0063  0020               ;
0064  0020               ;
0065  0020               ; .END
0065  0020               ;
0066  0020                      .LIB LCCVAR
```

```
LINE# LOC   CODE         LINE


0068  0020               ;
0069  0020               DRVST  *=*+2
0070  0022               DRVTRK *=*+2
0071  0024               STAB   *=*+10
0072  002E               ; VARIABLES
0073  002E               ;
0074  002E               ; POINTERS
0075  002E               SAVPNT *=*+2
0076  0030               BUFPNT *=*+2
0077  0032               HDRPNT *=*+2
0078  0034               ;
0079  0034               ;
0080  0034               GCRPNT *=*+1
0081  0035               GCRERR *=*+1           ; INDICATES GCR DECODE ERROR
0082  0036               BYTCNT *=*+1
0083  0037               BITCNT *=*+1
0084  0038               BID    *=*+1
0085  0039               HBID   *=*+1
0086  003A               CHKSUM *=*+1
0087  003B               HINIB  *=*+1
0088  003C               BYTE   *=*+1
0089  003D               DRIVE  *=*+1
0090  003E               CDRIVE *=*+1
0091  003F               JOBN   *=*+1
0092  0040               TRACC  *=*+1
0093  0041               NXTJOB *=*+1
0094  0042               NXTRK  *=*+1
0095  0043               SECTR  *=*+1
0096  0044               WORK   *=*+1
0097  0045               JOB    *=*+1
0098  0046               CTRACK *=*+1
0099  0047               DBID   *=*+1           ; DATA BLOCK ID
0100  0048               ACLTIM *=*+1           ; ACEL TIME DELAY
0101  0049               SAVSP  *=*+1           ; SAVE STACK POINTER
0102  004A               STEPS  *=*+1           ; STEPS TO DESIRED TRACK
0103  004B               TMP    *=*+1
0104  004C               CSECT  *=*+1
0105  004D               NEXTS  *=*+1
0106  004E               NXTBF  *=*+1           ; POINTER AT NEXT GCR SOURCE
                                                BUFFER
0107  004F               NXTPNT *=*+1           ; AND NEXT GCR BYTE LOCATION IN
                                                BUFFER
0108  0050               GCRFLG *=*+1           ; BUFFER IN GCR IMAGE
0109  0051               FTNUM  *=*+1           ; CURRENT FORMAT TRACK
0110  0052               BTAB   *=*+4
0111  0056               GTAB   *=*+8
0112  005E               ;
0113  005E               AS     *=*+1           ; # OF STEPS TO ACEL
0114  005F               AF     *=*+1           ; ACEL. FACTOR
0115  0060               ACLSTP *=*+1           ; STEPS TO GO
0116  0061               RSTEPS *=*+1           ; # OF RUN STEPS
0117  0062               NXTST  *=*+2
0118  0064               MINSTP *=*+1           ; MIN REQUIRED TO ACEL
0119  0065               ;
0120  0065               ;
0121  0065               ;
0122  0065               ; CONSTANTS
```

```
LINE# LOC   CODE        LINE


0123  0065              ;
0124  0065              OVRBUF =$0100         ; TOP OF STACK
0125  0065              NUMJOB =6             ; NUMBER OF JOBS
0126  0065              JMPC   =$50           ; JUMP COMMAND
0127  0065              BUMPC  =$40           ; BUMP COMMAND
0128  0065              EXECD  =$60           ; EXECUTE COMMAND
0129  0065              BUFS   =$0300         ; START OF BUFFERS
0130  0065              BUFF0  =BUFS
0131  0065              BUFF1  =BUFS+$100
0132  0065              BUFF2  =BUFS+$200
0133  0065              TOLONG =$2            ; FORMAT ERRORS
0134  0065              TOMANY =$3
0135  0065              TOBIG  =$4
0136  0065              TOSMAL =$5
0137  0065              NOTFND =$6
0138  0065              SKIP2  =$2C           ; BIT ABS
0139  0065              TOPRD  =69            ; TOP OF READ OVERFLO BUFFER ON A
                                              READ
0140  0065              TOPWRT =69            ; TOP OF WRITE OVERFLO BUFFER ON
                                              A WRITE
0141  0065              NUMSYN = 5            ; GCR BYTE COUNT FOR SIZE OF SYNC
                                              AREA
0142  0065              GAP1   = 10           ; GAP AFTER HEADER TO CLEAR ERASE
                                              IN GCR BYTES
0143  0065              ; 1541: GAP1 = 11
0144  0065              GAP2   = 4            ; GAP AFTER DATA BLOCK MIN SIZE
0145  0065              ; 1541: GAP2 = 2
0146  0065              RDMAX  = 6            ; SECTOR DISTANCE WAIT
0147  0065              WRTMIN = 9
0148  0065              WRTMAX = 12
0149  0065              TIM    = 58           ;IRQ RATE FOR 15 MS
0150  0065              ;
0151  0065              ;
0152  0065              ;
0153  0065              ;
0154  0065              ;
0155  0065              ;
0156  0065              ;.END
0156  0065              ;
0157  0065                     .LIB EQUATESF
```

```
LINE# LOC   CODE         LINE


0159  0065              ;******************************
0160  0065              ;* EQUATES
0161  0065              ;******************************
0162  0065              ;
0163  0065              ZP2    =*                     ;SAVE ZERO PAGE FOR DOS
0164  0065              ROM    =$C000                 ;FIRST ROM ADDRESS
0165  0065              ;
0166  0065              LRF    =$80                   ;LAST RECORD FLAG
0167  0065              DYFILE =$40                   ;DIRTY FLAG FOR RR FILE
0168  0065              OVRFLO =$20                   ;RR PRINT OVERFLOW
0169  0065              NSSL   =6                     ;# OF SIDE-SECTOR LINKS
0170  0065              SSIOFF =4+NSSL+NSSL           ;OFFSET INTO SS FOR DATA BLOCK
                                                       PTRS
0171  0065              NSSP   =120                   ;# OF PTRS IN SS
0172  0065              MXCHNS =6                     ;MAX # CHANNELS IN SYSTEM
0173  0065              MAXSA  =18                    ;MAX SA # +1
0174  0065              VERERR =7                     ;CONTROLLER VERIFY ERROR
0175  0065              CR     =$0D                   ; CARRIAGE RETURN
0176  0065              BFCNT  =5                     ;AVAILABLE BUFFER COUNT
0177  0065              CBPTR  =BFCNT+BFCNT           ;COMMAND BUFFER PTR
0178  0065              ERRCHN =MXCHNS-1              ;ERROR CHANNEL #
0179  0065              ERRSA  =16                    ;ERROR CHANNEL SA #
0180  0065              CMDCHN =MXCHNS-2              ;COMMAND CHANNEL #
0181  0065              LXINT  =%00001111             ;POWER UP LINUSE (LOGICAL INDEX
                                                       USAGE
0182  0065              BLINDX =6                     ;BAM LINDX FOR FLOATING BAMS
0183  0065              CMDSA  =15                    ;COMMAND CHANNEL SA #
0184  0065              APMODE =2                     ;OPEN APPEND MODE
0185  0065              MDMODE =3                     ;OPEN MODIFY MODE
0186  0065              RDMODE =0                     ;OPEN READ MODE
0187  0065              WTMODE =1                     ;OPEN WRITE MODE
0188  0065              RELTYP =4                     ;OPEN RELATIVE TYPE
0189  0065              DIRTYP =7                     ;OPEN DIRECT TYPE
0190  0065              SEQTYP =1                     ;OPEN SEQUENTIAL TYPE
0191  0065              PRGTYP =2                     ;OPEN PROGRAM TYPE
0192  0065              USRTYP =3                     ;OPEN USER TYPE
0193  0065              TYPMSK =7                     ;MASK FOR TYPE BITS
0194  0065              IRSA   =17                    ;INTERNAL READ SA #
0195  0065              IWSA   =18                    ;INTERNAL WRITE SA #
0196  0065              DOSVER =2                     ;DOS VERSION
0197  0065              FM2030 =$42                   ;2030 FORMAT VERSION
0198  0065              FM4040 =$41                   ;4040 FORMAT VERSION

0200  0065              ;CONTROLLER JOB TYPES
0201  0065              READ   =$80
0202  0065              WRITE  =$90
0203  0065              WVERFY =$A0
0204  0065              SEEK   =$B0
0205  0065              SECSEK =SEEK+8
0206  0065              BUMP   =$C0
0207  0065              JUMPC  =$D0
0208  0065              EXEC   =$E0
0209  0065
0210  0065              MXFILS =5                     ; MAX # FILENAMES IN STRING
0211  0065              DIRLEN =24                    ;DIRECTORY LENGTH USED
0212  0065              NBSIZ  =27                    ;NAMBUF TEXT SIZE
0213  0065              CMDLEN =41                    ;LENGTH OF COMMAND BUFFER
```

```
LINE# LOC   CODE        LINE


0215  0065              ;.END
0215  0065              ;
0216  0065                     .LIB IODEFSF
```

```
LINE# LOC   CODE         LINE


0218  0065              ;*****************************
0219  0065              ;* I/O DEFINITIONS           *
0220  0065              ;*****************************
0221  0065              ;
0222  0065              UNLSN  =$3F            ; IEEE UNLISTEN COMMAND
0223  0065              UNTLK  =$5F            ; IEEE UNTALK COMMAND
0224  0065              NOTRDY =$0             ; NOT READY
0225  0065              TALKER =$80            ; IEEE TALKER FLAG
0226  0065              LISNER =1              ; IEEE LISTENER FLAG
0227  0065              EOIOUT =$80            ; TALK WITH EOI
0228  0065              EOISND =$08            ; NOT(EOI) TO SEND
0229  0065              EOI    =$08            ; NOT(EOI) TO SEND
0230  0065              RDYTLK =$88            ; TALK NO EOI
0231  0065              RDYLST =$1             ; READY TO LISTEN
0232  0065              RNDRDY =RDYTLK+RDYLST  ; RANDOM CHNRDY
0233  0065              RNDEOI =EOIOUT+RDYLST  ; RANDOM W/ EOI


0235  0065              ;I/O REGISTERS

0237  0065              ; MOS 6522-A

0239  0065                     * =$1800
0240  1800              ;IEEE CONTROL PORT
0241  1800              PB     *=*+1           ; SERIAL PORT
0242  1801              PA1    *=*+1           ; IEEE DATA
0243  1802              DDRB1  *=*+1           ; SERIAL DATA DIR
0244  1803              DDRA1  *=*+1           ; IEEE DATA DIR
0245  1804              T1LC1  *=*+1           ; TIMER 1 LOW COUNTER
0246  1805              T1HC1  *=*+1           ; TIMER 1 HI COUNTER
0247  1806              T1LL1  *=*+1           ; TIMER 1 LOW LATCH
0248  1807              T1HL1  *=*+1           ; TIMER 1 HI LATCH
0249  1808              T2LC1  *=*+1           ; TIMER 2 LOW COUNTER
0250  1809              T2HC1  *=*+1           ; TIMER 2 HI COUNTER
0251  180A              SR1    *=*+1           ; SHIFT REG
0252  180B              ACR1   *=*+1           ; AUX CONTROL REG
0253  180C              PCR1   *=*+1           ;
0254  180D              IFR1   *=*+1           ;
0255  180E              IER1   *=*+1           ;
0256  180F              ;POTA1 *=*+1
0257  180F              IEEED  =PA1            ; IEEE  DATA PORT
0258  180F              ;
0259  180F              ;
0260  180F              ;
0261  180F              ; BITS FOR IEEE HANDSHAKE
0262  180F              ;
0263  180F              ATNA   =$1             ; ATN ACKNOWLEGDE
0264  180F              NRFD   =$2             ; NOT READY FOR DATA
0265  180F              NDAC   =$4             ; NOT DATA ACCEPTED
0266  180F              EOI    =$8             ; END OR IDENTIFY
0267  180F              TR     =$10            ; TRANSMISSION CONTROL FOR
                                               BUFFERS
0268  180F              DAV    =$40            ; DATA VALID
0269  180F              ATN    =$80            ; ATN IN
0270  180F              ;
```

```
LINE# LOC   CODE        LINE


0271  180F              ;
0272  180F              LED0   =8              ; ACT LED
0273  180F              LED1   =0              ; NO LED
0274  180F              LEDPRT=$1C00           ; ON PB OF $1C00
0275  180F              LEDOUT=$1C02           ; DDRB OF $1C00 FOR OUTPUT-LED
0276  180F              ;
0277  180F              ;
0278  180F              ;.END
0278  180F              ;
0279  180F                     .LIB RAMVARSF
```

```
LINE# LOC   CODE         LINE


0281  180F               ;PERMANENT ADDRESS VARIABLES
0282  180F               ;
0283  180F                      *=ZP2
0284  0065               ;
0285  0065               VNMI   *=*+2           ; INDIRECT FOR NMI
0286  0067               NMIFLG *=*+1
0287  0068               AUTOFG *=*+1
0288  0069               SECINC *=*+1           ; SECTOR INC FOR SEQ
0289  006A               REVCNT *=*+1           ; ERROR RECOVERY COUNT

0291  006B               BUFS   = $300          ; START OF DATA BUFS
0292  006B               FBUFS  = BUFS          ; FORMAT DOWNLOAD IMAGE

0294  006B               ;*
0295  006B               ;*********************************
0296  006B               ;*
0297  006B               ;* ZERO PAGE VARIABLES
0298  006B               ;*
0299  006B               ;*********************************
0300  006B               ;*
0301  006B               USRJMP *=*+2           ; USER JMP TABLE PTR
0302  006D               BMPNT  *=*+2           ; BIT MAP POINTER
0303  006F               TEMP   *=*+6           ; TEMP WORK SPACE
0304  0075               IP     *=*+2           ; INDIRECT PTR VARIABLE
0305  0077               LSNADR *=*+1           ; LISTEN ADDRESS
0306  0078               TLKADR *=*+1           ; TALKER ADDRESS
0307  0079               LSNACT *=*+1           ; ACTIVE LISTENER FLAG
0308  007A               TLKACT *=*+1           ; ACTIVE TALKER FLAG
0309  007B               ADRSED *=*+1           ; ADDRESSED FLAG
0310  007C               ATNPND *=*+1           ; ATTENTION PENDING FLAG
0311  007D               ATNMOD *=*+1           ; IN ATN MODE
0312  007E               PRGTRK *=*+1           ; LAST PROG ACCESSED
0313  007F               DRVNUM *=*+1           ; CURRENT DRIVE #
0314  0080               TRACK  *=*+1           ; CURRENT TRACK
0315  0081               SECTOR *=*+1           ; CURRENT SECTOR
0316  0082               LINDX  *=*+1           ; LOGICAL INDEX
0317  0083               SA     *=*+1           ; SECONDARY ADDRESS
0318  0084               ORGSA  *=*+1           ; ORIGINAL SA
0319  0085               DATA   *=*+1           ; TEMP DATA BYTE
0320  0086               ;*
0321  0086               ;*
0322  0086               T0     =TEMP
0323  0086               T1     =TEMP+1
0324  0086               T2     =TEMP+2
0325  0086               T3     =TEMP+3
0326  0086               T4     =TEMP+4
0327  0086               R0     *=*+1
0328  0087               R1     *=*+1
0329  0088               R2     *=*+1
0330  0089               R3     *=*+1
0331  008A               R4     *=*+1
0332  008B               RESULT *=*+4
0333  008F               ACCUM  *=*+5
0334  0094               DIRBUF *=*+2
```

```
LINE# LOC   CODE        LINE


0335  0096              ICMD   *=*+1           ; IEEE CMD IN
0336  0097              MYPA   *=*+1           ; MY PA FLAG
0337  0098              CONT   *=*+1           ; BIT COUNTER FOR SER
0338  0099              ;*
0339  0099              ;********************
0340  0099              ;*
0341  0099              ;* ZERO PAGE ARRAY
0342  0099              ;*
0343  0099              ;**********************
0344  0099              ;*
0345  0099              BUFTAB *=*+CBPTR+4     ; BUFFER BYTE POINTERS
0346  00A7              CB=BUFTAB+CBPTR
0347  00A7              BUF0   *=*+MXCHNS+1
0348  00AE              BUF1   *=*+MXCHNS+1
0349  00B5              NBKL
0350  00B5              RECL   *=*+MXCHNS
0351  00BB              NBKH
0352  00BB              RECH   *=*+MXCHNS
0353  00C1              NR     *=*+MXCHNS
0354  00C7              RS     *=*+MXCHNS
0355  00CD              SS     *=*+MXCHNS
0356  00D3              F1PTR  *=*+1           ; FILE STREAM 1 POINTER
0357  00D4              ;
0358  00D4              ;**********************
0359  00D4              ; $4300 VARS MOVED TO ZP
0360  00D4              ;
0361  00D4              RECPTR *=*+1
0362  00D5              SSNUM  *=*+1
0363  00D6              SSIND  *=*+1
0364  00D7              RELPTR *=*+1
0365  00D8              ENTSEC *=*+MXFILS      ; SECTOR OF DIRECTORY ENTRY
0366  00DD              ENTIND *=*+MXFILS      ; INDEX OF DIRECTORY ENTRY
0367  00E2              FILDRV *=*+MXFILS      ; DEFAULT FLAG, DRIVE #
0368  00E7              PATTYP *=*+MXFILS      ; PATTERN,REPLACE,CLOSED-
                                               FLAGS,TYPE
0369  00EC              FILTYP *=*+MXCHNS      ; CHANNEL FILE TYPE
0370  00F2              CHNRDY *=*+MXCHNS      ; CHANNEL STATUS
0371  00F8              EOIFLG *=*+1           ; TEMP EOI
0372  00F9              JOBNUM *=*+1           ; CURRENT JOB #
0373  00FA              LRUTBL *=*+MXCHNS-1    ; LEAST RECENTLY USED TABLE
0374  00FF              NODRV  *=*+2           ; NO DRIVE FLAG
0375  0101              DSKVER *=*+2           ; DISK VERSION FROM 18.0
0376  0103              ZPEND=*
0377  0103                     *=$200
0378  0200              CMDBUF *=*+CMDLEN+1
0379  022A              CMDNUM *=*+1           ; COMMAND #
0380  022B              LINTAB *=*+MAXSA+1     ; SA:LINDX TABLE
0381  023E              CHNDAT *=*+MXCHNS      ; CHANNEL DATA BYTE
0382  0244              LSTCHR *=*+MXCHNS      ; CHANNEL LAST CHAR PTR
0383  024A              TYPE   *=*+1           ; ACTIVE FILE TYPE
0384  024B              ;
0385  024B              ;*
0386  024B              ;******************
0387  024B              ;*
0388  024B              ;* RAM VARIABLES IN $4300
0389  024B              ;*
```

```
LINE# LOC   CODE         LINE


0390  024B               ;*******************
0391  024B               ;*
0392  024B               ; *=$4300
0393  024B               STRSIZ *=*+1
0394  024C               ;ZP: RECPTR *=*+1
0395  024C               ;ZP: SSNUM *=*+1
0396  024C               ;ZP: SSIND *=*+1
0397  024C               ;ZP: RELPTR *=*+1

0399  024C               TEMPSA *=*+1           ; TEMPORARY SA
0400  024D               ;ZP: EOIFLG *=*+1 ; TEMP EOI
0401  024D               CMD    *=*+1           ; TEMP JOB COMMAND

0403  024E               LSTSEC *=*+1           ;
0404  024F               BUFUSE *=*+2           ; BUFFER ALLOCATION
0405  0251               ;ZP: JOBNUM *=*+1 ; CURRENT JOB #
0406  0251               MDIRTY *=*+2           ; BAM 0 & 1 DIRTY FLAGS
0407  0253               ENTFND *=*+1           ; DIR-ENTRY FOUND FLAG
0408  0254               DIRLST *=*+1           ; DIR LISTING FLAG
0409  0255               CMDWAT *=*+1           ; COMMAND WAITING FLAG
0410  0256               LINUSE *=*+1           ; LINDX USE WORD
0411  0257               LBUSED *=*+1           ; LAST BUFFER USED
0412  0258               REC    *=*+1
0413  0259               TRKSS  *=*+1
0414  025A               SECSS  *=*+1
```

```
LINE# LOC   CODE        LINE


0416  025B              ;*
0417  025B              ;********************************
0418  025B              ;*
0419  025B              ;* RAM ARRAY AREA
0420  025B              ;*
0421  025B              ;********************************
0422  025B              ;*
0423  025B              LSTJOB *=*+BFCNT       ; LAST JOB
0424  0260              ;ZP: LINTAB *=*+MAXSA+1 ; SA:LINDX TABLE
0425  0260              ;ZP: CHNDAT *=*+MXCHNS ; CHANNEL DATA BYTE
0426  0260              DSEC   *=*+MXCHNS      ; SECTOR OF DIRECTORY ENTRY
0427  0266              DIND   *=*+MXCHNS      ; INDEX OF DIRECTORY ENTRY
0428  026C              ERWORD *=*+1           ; ERROR WORD FOR RECOVERY
0429  026D              ERLED  *=*+1           ; ERROR LED MASK FOR FLASHING
0430  026E              PRGDRV *=*+1           ; LAST PROGRAM DRIVE
0431  026F              PRGSEC *=*+1           ; LAST PROGRAM SECTOR
0432  0270              WLINDX *=*+1           ; WRITE LINDX
0433  0271              RLINDX *=*+1           ; READ LINDX
0434  0272              NBTEMP *=*+2           ; # BLOCKS TEMP


0436  0274              CMDSIZ *=*+1           ; COMMAND STRING SIZE
0437  0275              CHAR   *=*+1           ; CHAR UNDER PARSER
0438  0276              LIMIT  *=*+1           ; PTR LIMIT IN COMPAR
0439  0277              F1CNT  *=*+1           ; FILE STREAM 1 COUNT
0440  0278              F2CNT  *=*+1           ; FILE STREAM 2 COUNT
0441  0279              F2PTR  *=*+1           ; FILE STREAM 2 POINTER
0442  027A              ; PARSER TABLES
0443  027A              FILTBL *=*+MXFILS+1    ; FILENAME POINTER
0444  0280              ;ZP: FILENT *=*+MXFILS ; DIRECTORY ENTRY
0445  0280              ;ZP: FILDAT *=*+MXFILS ; DRIVE #, PATTERN
0446  0280              FILTRK *=*+MXFILS      ; 1ST LINK/TRACK
0447  0285              FILSEC *=*+MXFILS      ; /SECTOR

0449  028A              ; CHANNEL TABLES
0450  028A              ;ZP: FILTYP *=*+MXCHNS ; CHANNEL FILE TYPE
0451  028A              ;ZP: CHNRDY *=*+MXCHNS ; CHANNEL STATUS
0452  028A              ;ZP: LSTCHR *=*+MXCHNS ; CHANNEL LAST CHAR PTR

0454  028A              PATFLG *=*+1           ; PATTERN PRESENCE FLAG
0455  028B              IMAGE  *=*+1           ; FILE STREAM IMAGE
0456  028C              DRVCNT *=*+1           ; NUMBER OF DRV SEARCHES
0457  028D              DRVFLG *=*+1           ; DRIVE SEARCH FLAG
0458  028E              LSTDRV *=*+1           ; LAST DRIVE W/O ERROR
0459  028F              FOUND  *=*+1           ; FOUND FLAG IN DIR SEARCHES
0460  0290              DIRSEC *=*+1           ; DIRECTORY SECTOR
0461  0291              DELSEC *=*+1           ; SECTOR OF 1ST AVAIL ENTRY
0462  0292              DELIND *=*+1           ; INDEX "
0463  0293              LSTBUF *=*+1           ; =0 IF LAST BLOCK
0464  0294              INDEX  *=*+1           ; CURRENT INDEX IN BUFFER
0465  0295              FILCNT *=*+1           ; COUNTER, FILE ENTRIES
0466  0296              TYPFLG *=*+1           ; MATCH BY TYPE FLAG
0467  0297              MODE   *=*+1           ; ACTIVE FILE MODE (R,W)
0468  0298              ;ZP: TYPE *=*+1 ; ACTIVE FILE TYPE
```

```
LINE# LOC   CODE        LINE


0469  0298              JOBRTN *=*+1           ; JOB RETURN FLAG
0470  0299              EPTR   *=*+1           ; PTR FOR RECOVERY
0471  029A              TOFF   *=*+1           ; TOTAL TRACK OFFSET
0472  029B              UBAM   *=*+2           ; LAST BAM UPDATE PTR
0473  029D              TBAM   *=*+4           ; TRACK # OF BAM IMAGE
0474  02A1              BAM    *=*+16          ; BAM IMAGES

0476  02B1              ;*
0477  02B1              ;*****************************************
0478  02B1              ;*
0479  02B1              ;* OUTPUT BUFFERS
0480  02B1              ;*
0481  02B1              ;********************************************
0482  02B1              ;*
0483  02B1              ; *=$4400-36-36
0484  02B1              NAMBUF *=*+36          ; DIRECTORY BUFFER
0485  02D5              ERRBUF *=*+36          ; ERROR MSG BUFFER
0486  02F9              WBAM   *=*+1           ; DON'T-WRITE-BAM FLAG
0487  02FA              NDBL   *=*+2           ; # OF DISK BLOCKS FREE
0488  02FC              NDBH   *=*+2
0489  02FE              PHASE  *=*+2
0490  0300              RAMEND=*

0492  0300              ;.END
0492  0300              ;
0493  0300                     .LIB ROMSF
```

```
LINE# LOC   CODE         LINE


0495  0300                      *=ROM
0496  C000  B7           CCHKSM .BYTE $B7
0497  C001               ;
0498  C001               ;$C - $D PATCH AREA
0499  C001               ;----------------------------------------------
0500  C001               ; IN THE 2031 901484-03 ROM THE PATCH AREA
0501  C001               ; IS EMPTY
0502  C001               ;----------------------------------------------
0503  C001               ;
0504  C001               FREEC0 *=*+255         ; $C0 PATCH AREA
0505  C100               ;.END
0505  C100               ;
0506  C100                      .LIB LEDS
```

```
LINE# LOC   CODE        LINE


0508  C100              ;
0509  C100              ;TURN ON ACTIVITY LED SPECIFIED
0510  C100              ; BY DRVNUM
0511  C100              ;
0512  C100  78          SETLDS SEI
0513  C101  A9 F7              LDA #$FF-LED1-LED0
0514  C103  2D 00 1C           AND LEDPRT
0515  C106  48                 PHA
0516  C107              ;
0517  C107  A5 7F              LDA DRVNUM
0518  C109  F0 05              BEQ LEDS0
0519  C10B  68                 PLA
0520  C10C  09 00              ORA #LED1
0521  C10E  D0 03              BNE LEDS1
0522  C110              LEDS0
0523  C110  68                 PLA
0524  C111  09 08              ORA #LED0
0525  C113              LEDS1
0526  C113  8D 00 1C           STA LEDPRT
0527  C116  58                 CLI
0528  C117  60                 RTS
0529  C118              ;
0530  C118              ; SWITCH LED OF DRIVE 1(!) ON
0531  C118  78          LEDSON SEI
0532  C119  A9 08              LDA #LED1+LED0
0533  C11B  0D 00 1C           ORA LEDPRT
0534  C11E  8D 00 1C           STA LEDPRT
0535  C121  58                 CLI
0536  C122  60                 RTS
0537  C123              ;
0538  C123              ; CLEAR ERROR (STOPS LED BLINK)
0539  C123              ERROFF
0540  C123  A9 00              LDA #0
0541  C125  8D 6C 02           STA ERWORD
0542  C128  8D 6D 02           STA ERLED
0543  C12B  60                 RTS
0544  C12C              ;
0545  C12C              ; BLINK LED AFTER ERROR
0546  C12C  78          ERRON  SEI
0547  C12D  8A                 TXA
0548  C12E  48                 PHA            ; SAVE .X
0549  C12F  A9 50              LDA #80
0550  C131  8D 6C 02           STA ERWORD
0551  C134  A2 00              LDX #0
0552  C136              ; LDA DRVNUM ;FOR 2 DRIVES
0553  C136              ; AND #1
0554  C136              ; TAX
0555  C136  BD 19 FF           LDA LEDMSK,X
0556  C139  8D 6D 02           STA ERLED
0557  C13C  0D 00 1C           ORA LEDPRT     ; SET LED ON
0558  C13F  8D 00 1C           STA LEDPRT
0559  C142  68                 PLA
0560  C143  AA                 TAX            ; RESTORE .X
0561  C144  58                 CLI
0562  C145  60                 RTS
```

```
LINE# LOC   CODE        LINE


0563  C146              ; .END
0563  C146              ;
0564  C146                     .LIB PARSEX
```

```
LINE# LOC   CODE         LINE


C146              ;PARSE & EXECUTE STRING IN CMDBUF

C146              PARSXQ
C146  A9 00              LDA #0
C148  8D F9 02           STA WBAM
C14B  AD 8E 02           LDA LSTDRV
C14E  85 7F              STA DRVNUM
C150  20 BC E6           JSR OKERR
C153  A5 84              LDA ORGSA
C155  10 09              BPL PS05
C157  29 0F              AND #$F
C159  C9 0F              CMP #$F
C15B  F0 03              BEQ PS05
C15D  4C B4 D7           JMP OPEN
C160  20 B3 C2    PS05   JSR CMDSET       ; SET VARIABLES,REGS
C163  B1 A3              LDA (CB),Y
C165  8D 75 02           STA CHAR
C168  A2 0B              LDX #NCMDS-1    ; SEARCH CMD TABLE
C16A  BD D8 FE    PS10   LDA CMDTBL,X
C16D  CD 75 02           CMP CHAR
C170  F0 08              BEQ PS20
C172  CA                 DEX
C173  10 F5              BPL PS10
C175  A9 31              LDA #BADCMD     ; NO SUCH CMD
C177  4C C8 C1           JMP CMDERR
C17A  8E 2A 02    PS20   STX CMDNUM       ; X= CMD #

C17D  E0 09              CPX #PCMD       ; CMDS NOT PARSED
C17F  90 03              BCC PS30
C181  20 EE C1           JSR TAGCMD      ; SET TABLES, POINTERS &PATTERNS
C184  AE 2A 02    PS30   LDX CMDNUM
C187  BD E4 FE           LDA CJUMPL,X
C18A  85 6F              STA TEMP
C18C  BD F0 FE           LDA CJUMPH,X
C18F  85 70              STA TEMP+1
C191  6C 6F 00           JMP (TEMP)      ; COMMAND TABLE JUMP

C194              ; SUCCESSFUL COMMAND TERMINATION
C194              ENDCMD
C194  A9 00              LDA #0
C196  8D F9 02           STA WBAM
C199              ENDSAV
C199  AD 6C 02           LDA ERWORD
C19C  D0 2A              BNE CMDERR
C19E              ;
C19E  A0 00              LDY #0
C1A0  98                 TYA
C1A1  84 80              STY TRACK

C1A3  84 81       SCREND STY SECTOR       ; SCRATCH ENTRY
C1A5  84 A3              STY CB
C1A7  20 C7 E6           JSR ERRMSG
C1AA  20 23 C1           JSR ERROFF
C1AD              SCREN1
C1AD  A5 7F              LDA DRVNUM
```

```
LINE# LOC   CODE        LINE


0621  C1AF  8D 8E 02           STA LSTDRV
0622  C1B2  AA                 TAX
0623  C1B3  A9 00              LDA #0
0624  C1B5  95 FF              STA NODRV,X
0625  C1B7  20 BD C1           JSR CLRCB
0626  C1BA  4C DA D4           JMP FREICH     ; FREE INTERNAL CHANNEL
0627  C1BD              ;
0628  C1BD              CLRCB
0629  C1BD  A0 28              LDY #CMDLEN-1
0630  C1BF  A9 00              LDA #0
0631  C1C1              CLRB2
0632  C1C1  99 00 02           STA CMDBUF,Y
0633  C1C4  88                 DEY
0634  C1C5  10 FA              BPL CLRB2
0635  C1C7  60                 RTS
0636  C1C8              ;
0637  C1C8              ; COMMAND LEVEL ERROR PROCESSING
0638  C1C8  A0 00       CMDERR LDY #0
0639  C1CA  84 80              STY TRACK
0640  C1CC  84 81              STY SECTOR
0641  C1CE  4C 45 E6           JMP CMDER2

0643  C1D1              ; LOOK FOR A COLON IN THE CURRENT COMMAND, SWITCH LED OF
                                         DRIVE ON
0644  C1D1  A2 00       SIMPRS LDX #0           ; SIMPLE PARSER
0645  C1D3  8E 7A 02           STX FILTBL
0646  C1D6  A9 3A              LDA #':'
0647  C1D8  20 68 C2           JSR PARSE
0648  C1DB  F0 05              BEQ SP10
0649  C1DD  88                 DEY
0650  C1DE  88                 DEY
0651  C1DF  8C 7A 02           STY FILTBL
0652  C1E2  4C 68 C3    SP10   JMP SETANY       ; SET DRIVE #
```

```
LINE# LOC   CODE        LINE


0654  C1E5              PRSCLN
0655  C1E5  A0 00              LDY #0
0656  C1E7  A2 00              LDX #0
0657  C1E9  A9 3A              LDA #':'
0658  C1EB  4C 68 C2           JMP PARSE      ; FIND POS'N OF ":"
0659  C1EE              ;
0660  C1EE              ;TAG COMMAND STRING
0661  C1EE              ; SET UP CMD STRUCTURE
0662  C1EE              ; IMAGE & FILE STREAM PTRS
0663  C1EE              ;

0665  C1EE              TAGCMD
0666  C1EE  20 E5 C1           JSR PRSCLN
0667  C1F1  D0 05              BNE TC30
0668  C1F3  A9 34       TC25   LDA #NOFILE     ; NONE, NO FILES
0669  C1F5  4C C8 C1           JMP CMDERR
0670  C1F8  88          TC30   DEY
0671  C1F9  88                 DEY
0672  C1FA  8C 7A 02           STY FILTBL     ; ":"-1 STARTS FS1
0673  C1FD  8A                 TXA
0674  C1FE  D0 F3              BNE TC25       ; ERR: "," BEFORE ":"
0675  C200  A9 3D       TC35   LDA #'=         ; SEARCH: "="
0676  C202  20 68 C2           JSR PARSE
0677  C205  8A                 TXA            ; ?FILE COUNT= 1-1?
0678  C206  F0 02              BEQ TC40
0679  C208  A9 40              LDA #%01000000 ; G1-BIT
0680  C20A  09 21       TC40   ORA #%00100001  ; E1,.E2-BITS
0681  C20C  8D 8B 02           STA IMAGE      ; FS STRUCTURE
0682  C20F  E8                 INX
0683  C210  8E 77 02           STX F1CNT
0684  C213  8E 78 02           STX F2CNT      ; INIT FOR NO FS2
0685  C216  AD 8A 02           LDA PATFLG
0686  C219  F0 0D              BEQ TC50
0687  C21B  A9 80              LDA #%10000000 ; P1-BIT
0688  C21D  0D 8B 02           ORA IMAGE
0689  C220  8D 8B 02           STA IMAGE
0690  C223  A9 00              LDA #0
0691  C225  8D 8A 02           STA PATFLG     ; CLEAR PATTERN FLAG
0692  C228  98          TC50   TYA             ; PTR TO FS2
0693  C229  F0 29              BEQ TC75       ; FS2 NOT HERE
0694  C22B  9D 7A 02           STA FILTBL,X
0695  C22E  AD 77 02           LDA F1CNT      ; FS2 IS HERE NOW,...
0696  C231  8D 79 02           STA F2PTR      ; ...NOW SET F2 PTR
0697  C234  A9 8D              LDA #$8D       ; FIND CR-SHIFTED
0698  C236  20 68 C2           JSR PARSE      ; PARSE REST OF CMD STRING
0699  C239  E8                 INX            ; ADVANCE FILTBL PTR TO END
0700  C23A  8E 78 02           STX F2CNT      ; SAVE IT
0701  C23D  CA                 DEX            ; RESTORE FOR TEST
0702  C23E  AD 8A 02           LDA PATFLG     ; SAVE LAST PATTERN
0703  C241  F0 02              BEQ TC60       ; ?ANY PATTERNS?
0704  C243  A9 08              LDA #%1000     ; YES, P2-BIT
0705  C245  EC 77 02    TC60   CPX F1CNT       ; ?F2CNT=F1CNT+1?
0706  C248  F0 02              BEQ TC70
0707  C24A  09 04              ORA #%0100     ; G2-BIT
0708  C24C  09 03       TC70   ORA #%0011      ; E2-BIT,.E2-BIT
```

```
LINE# LOC   CODE        LINE


0709  C24E  4D 8B 02           EOR IMAGE      ; EOR CLEARS .E2-BIT
0710  C251  8D 8B 02           STA IMAGE
0711  C254              TC75
0712  C254  AD 8B 02           LDA IMAGE
0713  C257  AE 2A 02           LDX CMDNUM
0714  C25A  3D F4 FE           AND STRUCT,X   ; MATCH CMD TEMPLATE
0715  C25D  D0 01              BNE TC80
0716  C25F  60                 RTS
0717  C260  8D 6C 02    TC80   STA ERWORD      ; **COULD BE WARNING
0718  C263  A9 30              LDA #BADSYN    ; ERR: BAD SYNTAX
0719  C265  4C C8 C1           JMP CMDERR
```

```
LINE# LOC   CODE        LINE


0722  C268              ;PPARSE STRING
0723  C268              ; LOOKS FOR SPECIAL CHARS,
0724  C268              ; RETURNING WHEN VAR'BL CHAR
0725  C268              ; IS FOUND
0726  C268              ; A: VAR'BL CHAR
0727  C268              ; X: IN,OUT: INDEX, FILTBL+1
0728  C268              ; Y: IN: INDEX, CMDBUF
0729  C268              ; OUT: NEW PTR, =0 IF NONE
0730  C268              ; (Z=1) IF Y=0

0732  C268  8D 75 02    PARSE  STA CHAR        ; SAVE VAR'BL CHAR
0733  C26B  CC 74 02    PR10   CPY CMDSIZ      ; STAY IN STRING
0734  C26E  B0 2E              BCS PR30
0735  C270  B1 A3              LDA (CB),Y     ; MATCH CHAR
0736  C272  C8                 INY
0737  C273  CD 75 02           CMP CHAR
0738  C276  F0 28              BEQ PR35       ; FOUND CHAR
0739  C278  C9 2A              CMP #'*'       ; MATCH PATTERN CHARS
0740  C27A  F0 04              BEQ PR20
0741  C27C  C9 3F              CMP #'?'
0742  C27E  D0 03              BNE PR25
0743  C280  EE 8A 02    PR20   INC PATFLG      ; SET PATTERN FLAG
0744  C283  C9 2C       PR25   CMP #','        ; MATCH FILE SEPARATOR
0745  C285  D0 E4              BNE PR10
0746  C287  98                 TYA
0747  C288  9D 7B 02           STA FILTBL+1,X ; PUT PTRS IN TABLE
0748  C28B  AD 8A 02           LDA PATFLG     ; SAVE PATTERN FOR EA FILE
0749  C28E  29 7F              AND #$7F
0750  C290  F0 07              BEQ PR28
0751  C292  A9 80              LDA #$80       ; RETAIN PATTERN PRESENCE...
0752  C294  95 E7              STA PATTYP,X
0753  C296  8D 8A 02           STA PATFLG     ; ...BUT CLEAR COUNT
0754  C299  E8          PR28   INX
0755  C29A  E0 04              CPX #MXFILS-1
0756  C29C  90 CD              BCC PR10       ; NO MORE THAN MXFILS
0757  C29E  A0 00       PR30   LDY #0          ; Y=0 (Z=1)
0758  C2A0  AD 74 02    PR35   LDA CMDSIZ
0759  C2A3  9D 7B 02           STA FILTBL+1,X
0760  C2A6  AD 8A 02           LDA PATFLG
0761  C2A9  29 7F              AND #$7F
0762  C2AB  F0 04              BEQ PR40
0763  C2AD  A9 80              LDA #$80
0764  C2AF  95 E7              STA PATTYP,X
0765  C2B1  98          PR40   TYA             ; Z IS SET
0766  C2B2  60                 RTS

0768  C2B3              ;INITIALIZE COMMAND TABLES, PTRS, ETC.

0770  C2B3  A4 A3       CMDSET LDY BUFTAB+CBPTR
0771  C2B5  F0 14              BEQ CS08
0772  C2B7  88                 DEY
0773  C2B8  F0 10              BEQ CS07
0774  C2BA  B9 00 02           LDA CMDBUF,Y
0775  C2BD  C9 0D              CMP #CR
0776  C2BF  F0 0A              BEQ CS08
```

```
LINE# LOC   CODE        LINE


0777  C2C1  88                 DEY
0778  C2C2  B9 00 02           LDA CMDBUF,Y
0779  C2C5  C9 0D              CMP #CR
0780  C2C7  F0 02              BEQ CS08
0781  C2C9  C8                 INY
0782  C2CA  C8          CS07   INY
0783  C2CB  8C 74 02    CS08   STY CMDSIZ      ; SET CMD STRING SIZE

0785  C2CE  C0 2A              CPY #CMDLEN+1
0786  C2D0  A0 FF              LDY #$FF
0787  C2D2  90 08              BCC CMDRST
0788  C2D4  8C 2A 02           STY CMDNUM
0789  C2D7  A9 32              LDA #LONGLN    ; LONG LINE ERROR
0790  C2D9  4C C8 C1           JMP CMDERR
0791  C2DC              ; CLEAR VARIABLES,TABLES
0792  C2DC              CMDRST
0793  C2DC  A0 00              LDY #0
0794  C2DE  98                 TYA
0795  C2DF  85 A3              STA BUFTAB+CBPTR
0796  C2E1  8D 58 02           STA REC
0797  C2E4  8D 4A 02           STA TYPE
0798  C2E7  8D 96 02           STA TYPFLG
0799  C2EA  85 D3              STA F1PTR
0800  C2EC  8D 79 02           STA F2PTR
0801  C2EF  8D 77 02           STA F1CNT
0802  C2F2  8D 78 02           STA F2CNT
0803  C2F5  8D 8A 02           STA PATFLG
0804  C2F8  8D 6C 02           STA ERWORD
0805  C2FB  A2 05              LDX #MXFILS
0806  C2FD  9D 79 02    CS10   STA FILTBL-1,X
0807  C300  95 D7              STA ENTSEC-1,X
0808  C302  95 DC              STA ENTIND-1,X
0809  C304  95 E1              STA FILDRV-1,X
0810  C306  95 E6              STA PATTYP-1,X
0811  C308  9D 7F 02           STA FILTRK-1,X
0812  C30B  9D 84 02           STA FILSEC-1,X
0813  C30E  CA                 DEX
0814  C30F  D0 EC              BNE CS10
0815  C311  60                 RTS

0817  C312              ; .END
0817  C312              ;
0818  C312                     .LIB SETDRV
```

```
LINE# LOC   CODE        LINE


0820  C312              ;SET 1ST DRIVE AND TABLE POINTERS
0821  C312  AD 78 02    ONEDRV LDA F2CNT
0822  C315  8D 77 02           STA F1CNT
0823  C318  A9 01              LDA #1
0824  C31A  8D 78 02           STA F2CNT
0825  C31D  8D 79 02           STA F2PTR

0827  C320              ;SET UP ALL DRIVES FROM F2CNT
0828  C320  AC 8E 02    ALLDRS LDY LSTDRV       ; SET UP DRIVE #'S...
0829  C323  A2 00              LDX #0          ; ...INTO FILE ENTRY TABLE...
0830  C325  86 D3       AD10   STX F1PTR        ; ...ON SECTOR PTR BYTE
0831  C327  BD 7A 02           LDA FILTBL,X
0832  C32A  20 3C C3           JSR SETDRV
0833  C32D  A6 D3              LDX F1PTR
0834  C32F  9D 7A 02           STA FILTBL,X    ; INCR PTR PAST ":"
0835  C332  98                 TYA             ; BITS REP DRIVES
0836  C333  95 E2              STA FILDRV,X    ; BIT7: DEFAULT
0837  C335  E8                 INX             ; BIT0: DRIVE #
0838  C336  EC 78 02           CPX F2CNT
0839  C339  90 EA              BCC AD10
0840  C33B  60                 RTS

0842  C33C              ;SET DRIVE NUMBER
0843  C33C              ; DETERMINES DRIVE # FROM TEXT OR
0844  C33C              ; USES DEFAULT (-D)
0845  C33C              ; A: IN,OUT: INDEX, CMDBUF
0846  C33C              ; Y: IN: DEFAULT DRIVE!!!!!!!!!!!!!!!!!!!!!!!!!!!!
0847  C33C              ; OUT: DRIVE NUMBER, - IF DEFAULT

0849  C33C  AA          SETDRV TAX              ; X= CMDBUF INDEX
0850  C33D  A0 00              LDY #0          ; SET DEFAULT DRIVE TO
                                                ZERO!!!!!!!!!!!!!!!!!!!!!!
0851  C33F  A9 3A              LDA #':'
0852  C341  DD 01 02           CMP CMDBUF+1,X ; FOR XXX:FILE
0853  C344  F0 0C              BEQ SD40        ; .
0854  C346  DD 00 02           CMP CMDBUF,X    ; FOR XXX:FILE
0855  C349  D0 16              BNE SD50        ; .
0856  C34B  E8                 INX             ; FOUND ":", SO...
0857  C34C  98          SD20   TYA              ; DRIVE= DEFAULT
0858  C34D  29 01       SD22   AND #1           ; CONVERT TO NUMERIC
0859  C34F  A8          SD24   TAY              ; RESTORE DRIVE
0860  C350  8A                 TXA             ; A=INDEX & XXXXFILE
0861  C351  60                 RTS             ; .

0863  C352  BD 00 02    SD40   LDA CMDBUF,X
0864  C355  E8                 INX             ; XXX:FILE
0865  C356  E8                 INX             ; --.
0866  C357  C9 30              CMP #'0'        ; FOR XX0:FILE
0867  C359  F0 F2              BEQ SD22        ; .
0868  C35B  C9 31              CMP #'1'        ; FOR XX1:FILE
0869  C35D  F0 EE              BEQ SD22        ; .
0870  C35F  D0 EB              BNE SD20        ; CMD:FILE OR XX,:FILE
0871  C361              SD50   =*               ; . .
0872  C361  98                 TYA             ; FOR XXX,FILE OR XX=FILE
0873  C362  09 80              ORA #$80        ; . .
```

```
0874  C364  29 81              AND #$81       ; DRIVE= -DEFAULT
0875  C366  D0 E7              BNE SD24       ; FINISH TESTING (BRANCH ALWAYS)

0877  C368              ;SET CURRENT DRIVE NUMBER, SWITCH DRIVE LED ON
0878  C368  A9 00       SETANY LDA #0
0879  C36A  8D 8B 02           STA IMAGE
0880  C36D  AC 7A 02           LDY FILTBL
0881  C370  B1 A3       SA05   LDA (CB),Y
0882  C372  20 BD C3           JSR TST0V1
0883  C375  10 11              BPL SA20
0884  C377  C8                 INY
0885  C378  CC 74 02           CPY CMDSIZ
0886  C37B  B0 06              BCS SA10
0887  C37D  AC 74 02           LDY CMDSIZ
0888  C380  88                 DEY
0889  C381  D0 ED              BNE SA05
0890  C383  CE 8B 02    SA10   DEC IMAGE
0891  C386  A9 00              LDA #0         ;DEFAULT TO
                                               ZERO!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
                                               !!!!!
0892  C388  29 01       SA20   AND #1
0893  C38A  85 7F              STA DRVNUM
0894  C38C  4C 00 C1           JMP SETLDS
0895  C38F              ;TOGGLE DRVNUM
0896  C38F  A5 7F       TOGDRV LDA DRVNUM
0897  C391  49 01              EOR #1
0898  C393  29 01              AND #1
0899  C395  85 7F              STA DRVNUM
0900  C397  60                 RTS

0902  C398              ;SET PTRS TO ONE FILE STREAM & CHK TYPE
0903  C398  A0 00       FS1SET LDY #0
0904  C39A  AD 77 02           LDA F1CNT
0905  C39D  CD 78 02           CMP F2CNT
0906  C3A0  F0 16              BEQ FS15
0907  C3A2  CE 78 02           DEC F2CNT
0908  C3A5  AC 78 02           LDY F2CNT
0909  C3A8  B9 7A 02           LDA FILTBL,Y
0910  C3AB  A8                 TAY
0911  C3AC  B1 A3              LDA (CB),Y
0912  C3AE  A0 04              LDY #NTYPES-1
0913  C3B0  D9 0A FF    FS10   CMP TYPLST,Y
0914  C3B3  F0 03              BEQ FS15
0915  C3B5  88                 DEY
0916  C3B6  D0 F8              BNE FS10
0917  C3B8  98          FS15   TYA
0918  C3B9  8D 96 02           STA TYPFLG
0919  C3BC  60                 RTS

0921  C3BD              ;TEST CHAR IN ACCUM FOR "0" OR "1"
0922  C3BD  C9 30       TST0V1 CMP #'0'
0923  C3BF  F0 06              BEQ T0V1
0924  C3C1  C9 31              CMP #'1'
0925  C3C3  F0 02              BEQ T0V1
0926  C3C5  09 80              ORA #$80
0927  C3C7  29 81       T0V1   AND #$81
0928  C3C9  60                 RTS
```

```
LINE# LOC   CODE        LINE


0929  C3CA              ; .END
0929  C3CA              ;
0930  C3CA                     .LIB LOOKUP
```

```
LINE# LOC   CODE         LINE


0932  C3CA              ;OPTSCH OPTIMAL SEARCH FOR LOOKUP
0933  C3CA              ; AND FNDFIL

0935  C3CA  A9 00       OPTSCH LDA #0           ; DETERMINE OPTIMAL SEARCH
0936  C3CC  85 6F              STA TEMP        ; INIT DRIVE MASK
0937  C3CE  8D 8D 02           STA DRVFLG
0938  C3D1  48                 PHA
0939  C3D2  AE 78 02           LDX F2CNT
0940  C3D5  68          OS10   PLA
0941  C3D6  05 6F              ORA TEMP
0942  C3D8  48                 PHA
0943  C3D9  A9 01              LDA #1
0944  C3DB  85 6F              STA TEMP
0945  C3DD  CA                 DEX
0946  C3DE  30 0F              BMI OS30
0947  C3E0  B5 E2              LDA FILDRV,X
0948  C3E2  10 04              BPL OS15
0949  C3E4  06 6F              ASL TEMP
0950  C3E6  06 6F              ASL TEMP

0952  C3E8  4A          OS15   LSR A
0953  C3E9  90 EA              BCC OS10
0954  C3EB  06 6F              ASL TEMP
0955  C3ED  D0 E6              BNE OS10        ; (BRANCH)

0957  C3EF  68          OS30   PLA
0958  C3F0  AA                 TAX
0959  C3F1  BD 3F C4           LDA SCHTBL-1,X
0960  C3F4  48                 PHA
0961  C3F5  29 03              AND #3
0962  C3F7  8D 8C 02           STA DRVCNT
0963  C3FA  68                 PLA
0964  C3FB  0A                 ASL A
0965  C3FC  10 3E              BPL OS40
0966  C3FE  A5 E2              LDA FILDRV
0967  C400  29 01       OS35   AND #1
0968  C402  85 7F              STA DRVNUM
0969  C404              ;
0970  C404  AD 8C 02           LDA DRVCNT
0971  C407  F0 2B              BEQ OS60        ; ONLY ONE DRIVE ADDRESSED
0972  C409              ;
0973  C409  20 3D C6           JSR AUTOI       ; CHECK DRIVE FOR AUTOINIT
0974  C40C  F0 12              BEQ OS50        ; DRIVE IS ACTIVE
0975  C40E              ;
0976  C40E  20 8F C3           JSR TOGDRV
0977  C411  A9 00              LDA #0          ; SET 1 DRIVE ADDRESSED
0978  C413  8D 8C 02           STA DRVCNT
0979  C416  20 3D C6           JSR AUTOI       ; CHECK DRIVE FOR AUTOINIT
0980  C419  F0 1E              BEQ OS70        ; DRIVE IS ACTIVE
0981  C41B              OS45
0982  C41B  A9 74              LDA #NODRIV
0983  C41D  20 C8 C1           JSR CMDERR
0984  C420              OS50
0985  C420  20 8F C3           JSR TOGDRV
0986  C423  20 3D C6           JSR AUTOI       ; CHECK DRIVE FOR AUTOINIT
```

```
LINE# LOC   CODE        LINE


0987  C426  08                 PHP
0988  C427  20 8F C3           JSR TOGDRV
0989  C42A  28                 PLP
0990  C42B  F0 0C              BEQ OS70       ; DRIVE IS ACTIVE
0991  C42D              ;
0992  C42D  A9 00              LDA #0         ; SET 1 DRIVE ADDRESSED
0993  C42F  8D 8C 02           STA DRVCNT
0994  C432  F0 05              BEQ OS70       ; BRA
0995  C434              OS60
0996  C434  20 3D C6           JSR AUTOI      ; CHECK DRIVE FOR AUTOINIT
0997  C437  D0 E2              BNE OS45       ; DRIVE IS NOT ACTIVE
0998  C439              OS70
0999  C439  4C 00 C1           JMP SETLDS

1001  C43C  2A          OS40   ROL A
1002  C43D  4C 00 C4           JMP OS35

1004  C440  00          SCHTBL .BYTE $00, $80, $41
1004  C441  80
1004  C442  41
1005  C443  01                 .BYTE $01, $01, $01, $01
1005  C444  01
1005  C445  01
1005  C446  01
1006  C447  81                 .BYTE $81, $81, $81, $81
1006  C448  81
1006  C449  81
1006  C44A  81
1007  C44B  42                 .BYTE $42, $42, $42, $42
1007  C44C  42
1007  C44D  42
1007  C44E  42
```

```
LINE# LOC   CODE        LINE


1009  C44F              ; LOOK UP ALL FILES IN STREAM
1010  C44F              ; AND FILL TABLES W/ INFO

1012  C44F  20 CA C3    LOOKUP JSR OPTSCH
1013  C452  A9 00       LK05   LDA #0
1014  C454  8D 92 02           STA DELIND
1015  C457  20 AC C5           JSR SRCHST     ; START SEARCH
1016  C45A  D0 19              BNE LK25
1017  C45C  CE 8C 02    LK10   DEC DRVCNT
1018  C45F  10 01              BPL LK15
1019  C461  60                 RTS            ; NO MORE DRIVE SEARCHES
1020  C462  A9 01       LK15   LDA #1          ; TOGGLE DRIVE #
1021  C464  8D 8D 02           STA DRVFLG
1022  C467  20 8F C3           JSR TOGDRV
1023  C46A  20 00 C1           JSR SETLDS     ; TURN ON LED
1024  C46D  4C 52 C4           JMP LK05
1025  C470  20 17 C6    LK20   JSR SEARCH      ; FIND VALID FN
1026  C473  F0 10              BEQ LK30       ; END OF SEARCH
1027  C475  20 D8 C4    LK25   JSR COMPAR      ; COMPARE DIR W/ TABLE
1028  C478  AD 8F 02           LDA FOUND      ; FOUND FLAG
1029  C47B  F0 01              BEQ LK26       ; ALL FN'S NOT FOUND, YET
1030  C47D  60                 RTS

1032  C47E  AD 53 02    LK26   LDA ENTFND
1033  C481  30 ED              BMI LK20
1034  C483  10 F0              BPL LK25

1036  C485  AD 8F 02    LK30   LDA FOUND
1037  C488  F0 D2              BEQ LK10
1038  C48A  60                 RTS

1040  C48B              ; FIND NEXT FILE NAME MATCHING
1041  C48B              ; ANY FILE IN STREAM & RETURN
1042  C48B              ; WITH ENTRY FOUND STUFFED INTO
1043  C48B              ; TABLES
1044  C48B  20 04 C6    FFRE   JSR SRRE        ; FIND FILE RE-ENTRY
1045  C48E  F0 1A              BEQ FF10
1046  C490  D0 28              BNE FF25

1048  C492  A9 01       FF15   LDA #1
1049  C494  8D 8D 02           STA DRVFLG
1050  C497  20 8F C3           JSR TOGDRV
1051  C49A  20 00 C1           JSR SETLDS

1053  C49D  A9 00       FFST   LDA #0          ; FIND FILE START ENTRY
1054  C49F  8D 92 02           STA DELIND
1055  C4A2  20 AC C5           JSR SRCHST
1056  C4A5  D0 13              BNE FF25
1057  C4A7  8D 8F 02           STA FOUND
1058  C4AA  AD 8F 02    FF10   LDA FOUND
1059  C4AD  D0 28              BNE FF40
1060  C4AF  CE 8C 02           DEC DRVCNT
1061  C4B2  10 DE              BPL FF15
1062  C4B4  60                 RTS
```

```
LINE# LOC   CODE        LINE


1064  C4B5  20 17 C6    FNDFIL JSR SEARCH      ; FIND FILE CONTINUOUS...
1065  C4B8  F0 F0              BEQ FF10       ; ... RE-ENTRY, NO CHANNEL
                                                ACTIVITY
1066  C4BA  20 D8 C4    FF25   JSR COMPAR      ; COMPARE FILE NAMES
1067  C4BD  AE 53 02           LDX ENTFND
1068  C4C0  10 07              BPL FF30
1069  C4C2  AD 8F 02           LDA FOUND
1070  C4C5  F0 EE              BEQ FNDFIL
1071  C4C7  D0 0E              BNE FF40

1073  C4C9  AD 96 02    FF30   LDA TYPFLG
1074  C4CC  F0 09              BEQ FF40       ; NO TYPE RESTRICTION
1075  C4CE  B5 E7              LDA PATTYP,X
1076  C4D0  29 07              AND #TYPMSK
1077  C4D2  CD 96 02           CMP TYPFLG
1078  C4D5  D0 DE              BNE FNDFIL
1079  C4D7  60          FF40   RTS
```

```
LINE# LOC   CODE        LINE


1081  C4D8              ;COMPARE ALL FILENAMES IN STREAM TABLE
1082  C4D8              ; WITH EACH VALID ENTRY IN THE
1083  C4D8              ; DIRECTORY. MATCHES ARE TABULATED

1085  C4D8  A2 FF       COMPAR LDX #$FF
1086  C4DA  8E 53 02           STX ENTFND
1087  C4DD  E8                 INX
1088  C4DE  8E 8A 02           STX PATFLG
1089  C4E1  20 89 C5           JSR CMPCHK
1090  C4E4  F0 06              BEQ CP10
1091  C4E6  60          CP02   RTS              ; ALL ARE FOUND

1093  C4E7  20 94 C5    CP05   JSR CC10
1094  C4EA  D0 FA              BNE CP02
1095  C4EC  A5 7F       CP10   LDA DRVNUM
1096  C4EE  55 E2              EOR FILDRV,X
1097  C4F0  4A                 LSR A
1098  C4F1  90 0B              BCC CP20       ; RIGHT DRIVE
1099  C4F3  29 40              AND #$40
1100  C4F5  F0 F0              BEQ CP05       ; NO DEFAULT
1101  C4F7  A9 02              LDA #2
1102  C4F9  CD 8C 02           CMP DRVCNT
1103  C4FC  F0 E9              BEQ CP05       ; DON'T USE DEFAULT

1105  C4FE  BD 7A 02    CP20   LDA FILTBL,X     ; GOOD DRIVE MATCH
1106  C501  AA                 TAX
1107  C502  20 A6 C6           JSR FNDLMT
1108  C505  A0 03              LDY #3
1109  C507  4C 1D C5           JMP CP33
1110  C50A              CP30
1111  C50A  BD 00 02           LDA CMDBUF,X
1112  C50D  D1 94              CMP (DIRBUF),Y
1113  C50F  F0 0A              BEQ CP32       ; CHARS ARE =
1114  C511              ;
1115  C511  C9 3F              CMP #'?
1116  C513  D0 D2              BNE CP05       ; NO SINGLE PATTERN
1117  C515  B1 94              LDA (DIRBUF),Y
1118  C517  C9 A0              CMP #$A0
1119  C519  F0 CC              BEQ CP05       ; END OF FILENAME
1120  C51B              CP32
1121  C51B  E8                 INX
1122  C51C  C8                 INY
1123  C51D              CP33
1124  C51D  EC 76 02           CPX LIMIT
1125  C520  B0 09              BCS CP34       ; END OF PATTERN
1126  C522              ;
1127  C522  BD 00 02           LDA CMDBUF,X
1128  C525  C9 2A              CMP #'*
1129  C527  F0 0C              BEQ CP40       ; STAR MATCHES ALL
1130  C529  D0 DF              BNE CP30       ; KEEP CHECKING
1131  C52B              CP34
1132  C52B  C0 13              CPY #19
1133  C52D  B0 06              BCS CP40       ; END OF FILENAME
1134  C52F              ;
1135  C52F  B1 94              LDA (DIRBUF),Y
```

```
LINE# LOC   CODE         LINE


C531  C9 A0              CMP #$A0
C533  D0 B2              BNE CP05

C535  AE 79 02    CP40   LDX F2PTR       ; FILENAMES MATCH
C538  8E 53 02           STX ENTFND
C53B  B5 E7              LDA PATTYP,X   ; STORE INFO IN TABLES
C53D  29 80              AND #$80
C53F  8D 8A 02           STA PATFLG
C542  AD 94 02           LDA INDEX
C545  95 DD              STA ENTIND,X
C547  A5 81              LDA SECTOR
C549  95 D8              STA ENTSEC,X
C54B  A0 00              LDY #0
C54D  B1 94              LDA (DIRBUF),Y
C54F  C8                 INY
C550  48                 PHA
C551  29 40              AND #$40
C553  85 6F              STA TEMP
C555  68                 PLA
C556  29 DF              AND #$FF-$20
C558  30 02              BMI CP42
C55A              ;
C55A  09 20              ORA #$20
C55C              CP42
C55C  29 27              AND #$27
C55E  05 6F              ORA TEMP
C560  85 6F              STA TEMP
C562  A9 80              LDA #$80
C564  35 E7              AND PATTYP,X
C566  05 6F              ORA TEMP
C568  95 E7              STA PATTYP,X
C56A  B5 E2              LDA FILDRV,X
C56C  29 80              AND #$80
C56E  05 7F              ORA DRVNUM
C570  95 E2              STA FILDRV,X
C572              ;
C572  B1 94              LDA (DIRBUF),Y
C574  9D 80 02           STA FILTRK,X
C577  C8                 INY
C578  B1 94              LDA (DIRBUF),Y
C57A  9D 85 02           STA FILSEC,X
C57D  AD 58 02           LDA REC
C580  D0 07              BNE CP50
C582  A0 15              LDY #21
C584  B1 94              LDA (DIRBUF),Y
C586  8D 58 02           STA REC
C589              CP50
C589              ; JMP CMPCHK
C589              ; RTS

C589              ;CHECK TABLE FOR UNFOUND FILES

C589  A9 FF       CMPCHK LDA #$FF
C58B  8D 8F 02           STA FOUND
C58E  AD 78 02           LDA F2CNT
```

```
LINE# LOC   CODE        LINE


1191  C591  8D 79 02           STA F2PTR

1193  C594  CE 79 02    CC10   DEC F2PTR
1194  C597  10 01              BPL CC15
1195  C599  60                 RTS             ; TABLE EXHAUSTED

1197  C59A  AE 79 02    CC15   LDX F2PTR
1198  C59D  B5 E7              LDA PATTYP,X
1199  C59F  30 05              BMI CC20
1200  C5A1  BD 80 02           LDA FILTRK,X
1201  C5A4  D0 EE              BNE CC10
1202  C5A6  A9 00       CC20   LDA #0
1203  C5A8  8D 8F 02           STA FOUND
1204  C5AB  60                 RTS
```

```
LINE# LOC   CODE        LINE


1206  C5AC              ;SEARCH DIRECTORY
1207  C5AC              ; RETURNS WITH VALID ENTRY W/ DELIND=0
1208  C5AC              ; OR RETURNS W/ 1ST DELETED ENTRY
1209  C5AC              ; W/ DELIND=1
1210  C5AC              ;
1211  C5AC              ; SRCHST WILL INITIATE A SEARCH
1212  C5AC              ; SEARCH WILL CONTINUE A SEARCH

1214  C5AC              SRCHST
1215  C5AC  A0 00              LDY #0          ; INIT DELETED SECTOR
1216  C5AE  8C 91 02           STY DELSEC
1217  C5B1  88                 DEY
1218  C5B2  8C 53 02           STY ENTFND

1220  C5B5  AD D4 FE           LDA DIRTRK      ; START SEARCH AT BEGINNING
1221  C5B8  85 80              STA TRACK
1222  C5BA  A9 01              LDA #1
1223  C5BC  85 81              STA SECTOR
1224  C5BE  8D 93 02           STA LSTBUF
1225  C5C1  20 75 D4           JSR OPNIRD      ; OPEN INTERNAL READ CHNL

1227  C5C4  AD 93 02    SR10   LDA LSTBUF       ; LAST BUFFER IF 0
1228  C5C7  D0 01              BNE SR15
1229  C5C9  60                 RTS             ; (Z=1)

1231  C5CA  A9 07       SR15   LDA #7
1232  C5CC  8D 95 02           STA FILCNT
1233  C5CF  A9 00              LDA #0          ; READ TRACK #
1234  C5D1  20 F6 D4           JSR DRDBYT
1235  C5D4  8D 93 02           STA LSTBUF      ; UPDATE END FLAG

1237  C5D7  20 E8 D4    SR20   JSR GETPNT
1238  C5DA  CE 95 02           DEC FILCNT
1239  C5DD  A0 00              LDY #0
1240  C5DF  B1 94              LDA (DIRBUF),Y ; READ FILE TYPE
1241  C5E1  D0 18              BNE SR30

1243  C5E3  AD 91 02           LDA DELSEC      ; DELETED ENTRY FOUND
1244  C5E6  D0 2F              BNE SEARCH      ; DELETED ENTRY ALREADY FOUND
1245  C5E8  20 3B DE           JSR CURBLK      ; GET CURRENT SECTOR
1246  C5EB  A5 81              LDA SECTOR
1247  C5ED  8D 91 02           STA DELSEC

1249  C5F0  A5 94              LDA DIRBUF      ; GET CURRENT INDEX
1250  C5F2  AE 92 02           LDX DELIND      ; BIT1: WANT DELETED ENTRY
1251  C5F5  8D 92 02           STA DELIND
1252  C5F8  F0 1D              BEQ SEARCH      ; NEED VALID ENTRY
1253  C5FA  60                 RTS             ; (Z=0)
1254  C5FB  A2 01       SR30   LDX #1
1255  C5FD  EC 92 02           CPX DELIND      ; ?LOOKING FOR DELETED?
1256  C600  D0 2D              BNE SR50        ; NO!
1257  C602  F0 13              BEQ SEARCH

1259  C604  AD D4 FE    SRRE   LDA DIRTRK
1260  C607  85 80              STA TRACK
```

```
LINE# LOC   CODE        LINE


1261  C609  AD 90 02           LDA DIRSEC
1262  C60C  85 81              STA SECTOR
1263  C60E  20 75 D4           JSR OPNIRD
1264  C611  AD 94 02           LDA INDEX
1265  C614  20 C8 D4           JSR SETPNT

1267  C617  A9 FF       SEARCH LDA #$FF
1268  C619  8D 53 02           STA ENTFND
1269  C61C  AD 95 02           LDA FILCNT     ; ADJUST FILE COUNT
1270  C61F  30 08              BMI SR40
1271  C621  A9 20              LDA #32        ; INCR BY 32
1272  C623  20 C6 D1           JSR INCPTR
1273  C626  4C D7 C5           JMP SR20

1275  C629  20 4D D4    SR40   JSR NXTBUF      ; NEW BUFFER
1276  C62C  4C C4 C5           JMP SR10       ; (BRANCH)

1278  C62F  A5 94       SR50   LDA DIRBUF      ; FOUND VALID ENTRY
1279  C631  8D 94 02           STA INDEX      ; SAVE INDEX
1280  C634  20 3B DE           JSR CURBLK     ; GET SECTOR
1281  C637  A5 81              LDA SECTOR
1282  C639  8D 90 02           STA DIRSEC

1284  C63C  60                 RTS            ; (Z=0)
1285  C63D              AUTOI
1286  C63D              ; CHECK DRIVE FOR ACTIVE DISKETTE
1287  C63D              ; INIT IF NEEDED
1288  C63D              ; RETURN NODRV STATUS
1289  C63D              ;
1290  C63D  A5 68              LDA AUTOFG
1291  C63F  D0 28              BNE AUTO2      ; AUTO-INIT IS DISABLED
1292  C641              ;
1293  C641  A6 7F              LDX DRVNUM
1294  C643  56 1C              LSR WPSW,X     ; TEST & CLEAR WPSW
1295  C645  90 22              BCC AUTO2      ; NO CHANGE IN DISKETTE
1296  C647              ;
1297  C647  A9 FF              LDA #$FF
1298  C649  8D 98 02           STA JOBRTN     ; SET ERROR RETURN CODE
1299  C64C  20 0E D0           JSR ITRIAL     ; INIT-SEEK TEST
1300  C64F  A0 FF              LDY #$FF       ; .Y= TRUE
1301  C651  C9 02              CMP #2
1302  C653  F0 0A              BEQ AUTO1      ; NO SYNC= NO DISKETTE
1303  C655              ;
1304  C655  C9 03              CMP #3
1305  C657  F0 06              BEQ AUTO1      ; NO HEADER= NO DIRECTORY
1306  C659              ;
1307  C659  C9 0F              CMP #$F
1308  C65B  F0 02              BEQ AUTO1      ; NO DRIVE!!!!
1309  C65D              ;
1310  C65D  A0 00              LDY #0         ; SET .Y FALSE
1311  C65F              AUTO1
1312  C65F  A6 7F              LDX DRVNUM
1313  C661  98                 TYA
1314  C662  95 FF              STA NODRV,X    ; SET CONDN OF NO-DRIVE
1315  C664  D0 03              BNE AUTO2      ; NO NEED TO INIT CRUD!
```

```
LINE# LOC   CODE         LINE


1316  C666               ;
1317  C666  20 42 D0            JSR INITDR     ; INIT THAT DRIVE
1318  C669  A6 7F        AUTO2  LDX DRVNUM
1319  C66B  B5 FF               LDA NODRV,X
1320  C66D  60                  RTS

1322  C66E               ; .END
1322  C66E               ;
1323  C66E                      .LIB TRNSFR
```

```
LINE# LOC   CODE        LINE


1325  C66E              ;TRANSFER FILENAME FROM CMD TO BUFFER
1326  C66E              ; A: STRING SIZE
1327  C66E              ; X: STARTING INDEX IN CMDBUF
1328  C66E              ; Y: BUFFER #
1329  C66E  48          TRNAME PHA
1330  C66F  20 A6 C6           JSR FNDLMT
1331  C672  20 88 C6           JSR TRCMBF
1332  C675  68                 PLA
1333  C676  38                 SEC
1334  C677  ED 4B 02           SBC STRSIZ
1335  C67A  AA                 TAX
1336  C67B  F0 0A              BEQ TN20
1337  C67D  90 08              BCC TN20

1339  C67F  A9 A0              LDA #$A0
1340  C681  91 94       TN10   STA (DIRBUF),Y
1341  C683  C8                 INY
1342  C684  CA                 DEX
1343  C685  D0 FA              BNE TN10
1344  C687  60          TN20   RTS

1346  C688              ;TRANSFER CMD BUFFER TO OTHER BUFFER
1347  C688              ; USES CURRENT BUFFER PTR
1348  C688              ; LIMIT: ENDING INDEX+1 IN CMD BUF
1349  C688              ; X: STARTING INDEX IN CMD BUF
1350  C688              ; Y: BUFFER #

1352  C688  98          TRCMBF TYA
1353  C689  0A                 ASL A
1354  C68A  A8                 TAY
1355  C68B  B9 99 00           LDA BUFTAB,Y
1356  C68E  85 94              STA DIRBUF
1357  C690  B9 9A 00           LDA BUFTAB+1,Y
1358  C693  85 95              STA DIRBUF+1
1359  C695  A0 00              LDY #0
1360  C697              ;
1361  C697  BD 00 02    TR10   LDA CMDBUF,X
1362  C69A  91 94              STA (DIRBUF),Y
1363  C69C  C8                 INY
1364  C69D  F0 06              BEQ TR20
1365  C69F  E8                 INX
1366  C6A0  EC 76 02           CPX LIMIT
1367  C6A3  90 F2              BCC TR10
1368  C6A5  60          TR20   RTS
1369  C6A6              ;
1370  C6A6              ;
1371  C6A6              ;FIND THE LIMIT OF THE STRING IN CMDBUF
1372  C6A6              ; POINTED TO BY X

1374  C6A6  A9 00       FNDLMT LDA #0
1375  C6A8  8D 4B 02           STA STRSIZ
1376  C6AB  8A                 TXA
1377  C6AC  48                 PHA
1378  C6AD              FL05
1379  C6AD  BD 00 02           LDA CMDBUF,X
```

```
LINE# LOC   CODE        LINE


1380  C6B0  C9 2C              CMP #','
1381  C6B2  F0 14              BEQ FL10
1382  C6B4  C9 3D              CMP #'='
1383  C6B6  F0 10              BEQ FL10
1384  C6B8  EE 4B 02           INC STRSIZ
1385  C6BB  E8                 INX
1386  C6BC  A9 0F              LDA #15
1387  C6BE  CD 4B 02           CMP STRSIZ
1388  C6C1  90 05              BCC FL10
1389  C6C3  EC 74 02           CPX CMDSIZ
1390  C6C6  90 E5              BCC FL05
1391  C6C8  8E 76 02    FL10   STX LIMIT
1392  C6CB  68                 PLA
1393  C6CC  AA                 TAX
1394  C6CD  60                 RTS
```

```
LINE# LOC   CODE         LINE


1397  C6CE              ; GET FILE ENTRY FROM DIRECTORY
1398  C6CE              ; CALLED BY STDIR, GETDIR
1399  C6CE  A5 83       GETNAM LDA SA           ;SAVE VARIABLES
1400  C6D0  48                 PHA
1401  C6D1  A5 82              LDA LINDX
1402  C6D3  48                 PHA
1403  C6D4  20 DE C6           JSR GNSUB
1404  C6D7  68                 PLA             ;RESTORE VARIABLES
1405  C6D8  85 82              STA LINDX
1406  C6DA  68                 PLA
1407  C6DB  85 83              STA SA
1408  C6DD  60                 RTS
1409  C6DE              ;
1410  C6DE  A9 11       GNSUB  LDA #IRSA
1411  C6E0  85 83              STA SA
1412  C6E2  20 EB D0           JSR FNDRCH
1413  C6E5  20 E8 D4           JSR GETPNT
1414  C6E8  AD 53 02           LDA ENTFND
1415  C6EB  10 0A              BPL GN05        ;MORE FILES
1416  C6ED  AD 8D 02           LDA DRVFLG
1417  C6F0  D0 0A              BNE GN050
1418  C6F2  20 06 C8           JSR MSGFRE      ; SEND BLOCKS FREE
1419  C6F5  18                 CLC             ; (C=0): END
1420  C6F6  60                 RTS             ; TERMINATE

1422  C6F7  AD 8D 02    GN05   LDA DRVFLG       ;(DRVFLG=0):
1423  C6FA  F0 1F              BEQ GN10        ;SEND FILE NAME
1424  C6FC              GN050
1425  C6FC  CE 8D 02           DEC DRVFLG      ;(DRVFLG=-1):NEW DIR
1426  C6FF  D0 0D              BNE GN051
1427  C701  CE 8D 02           DEC DRVFLG      ;(DRVFLG=-1):
1428  C704  20 8F C3           JSR TOGDRV      ; SEND BLOCKS FREE
1429  C707  20 06 C8           JSR MSGFRE
1430  C70A  38                 SEC
1431  C70B  4C 8F C3           JMP TOGDRV

1433  C70E  A9 00       GN051  LDA #0
1434  C710  8D 73 02           STA NBTEMP+1
1435  C713  8D 8D 02           STA DRVFLG      ;RESET FLAG
1436  C716  20 B7 C7           JSR NEWDIR
1437  C719  38                 SEC
1438  C71A  60                 RTS

1440  C71B  A2 18       GN10   LDX #DIRLEN      ;SET NUMBER BLOCKS
1441  C71D  A0 1D              LDY #29         ; & ADJUST SPACING
1442  C71F  B1 94              LDA (DIRBUF),Y
1443  C721  8D 73 02           STA NBTEMP+1
1444  C724  F0 02              BEQ GN12
1445  C726  A2 16              LDX #DIRLEN-2
1446  C728  88          GN12   DEY
1447  C729  B1 94              LDA (DIRBUF),Y
1448  C72B  8D 72 02           STA NBTEMP
1449  C72E  E0 16              CPX #DIRLEN-2
1450  C730  F0 0A              BEQ GN14
1451  C732  C9 0A              CMP #10
```

```
LINE# LOC   CODE        LINE


1452  C734  90 06              BCC GN14
1453  C736  CA                 DEX
1454  C737  C9 64              CMP #100
1455  C739  90 01              BCC GN14
1456  C73B  CA                 DEX

1458  C73C  20 AC C7    GN14   JSR BLKNB       ;CLEAR NAME BUFFER
1459  C73F              ;
1460  C73F  B1 94              LDA (DIRBUF),Y ;SET TYPE CHARS
1461  C741  48                 PHA
1462  C742  0A                 ASL A          ;(USED IN BCS)
1463  C743  10 05              BPL GN15
1464  C745  A9 3C              LDA #'<'
1465  C747  9D B2 02           STA NAMBUF+1,X
1466  C74A              GN15
1467  C74A  68                 PLA
1468  C74B  29 0F              AND #$F
1469  C74D  A8                 TAY
1470  C74E  B9 14 FF           LDA TP2LST,Y
1471  C751  9D B1 02           STA NAMBUF,X
1472  C754  CA                 DEX
1473  C755  B9 0F FF           LDA TP1LST,Y
1474  C758  9D B1 02           STA NAMBUF,X
1475  C75B  CA                 DEX
1476  C75C  B9 0A FF           LDA TYPLST,Y
1477  C75F  9D B1 02           STA NAMBUF,X
1478  C762  CA                 DEX
1479  C763  CA                 DEX
1480  C764  B0 05              BCS GN20       ;(FROM ASL)
1481  C766  A9 2A              LDA #'*'       ;FILE NOT CLOSED
1482  C768  9D B2 02           STA NAMBUF+1,X
1483  C76B  A9 A0       GN20   LDA #$A0
1484  C76D  9D B1 02           STA NAMBUF,X
1485  C770  CA                 DEX
1486  C771  A0 12              LDY #18
1487  C773  B1 94       GN22   LDA (DIRBUF),Y
1488  C775  9D B1 02           STA NAMBUF,X
1489  C778  CA                 DEX
1490  C779  88                 DEY
1491  C77A  C0 03              CPY #3
1492  C77C  B0 F5              BCS GN22

1494  C77E  A9 22              LDA #'"        ;SEND NAME IN QUOTES
1495  C780  9D B1 02           STA NAMBUF,X
1496  C783  E8          GN30   INX
1497  C784  E0 20              CPX #$20
1498  C786  B0 0B              BCS GN35
1499  C788  BD B1 02           LDA NAMBUF,X
1500  C78B  C9 22              CMP #'"'
1501  C78D  F0 04              BEQ GN35
1502  C78F  C9 A0              CMP #$A0
1503  C791  D0 F0              BNE GN30
1504  C793  A9 22       GN35   LDA #'"'
1505  C795  9D B1 02           STA NAMBUF,X
1506  C798  E8          GN37   INX
```

```
LINE# LOC   CODE        LINE


1507  C799  E0 20              CPX #$20
1508  C79B  B0 0A              BCS GN40
1509  C79D  A9 7F              LDA #$7F
1510  C79F  3D B1 02           AND NAMBUF,X
1511  C7A2  9D B1 02           STA NAMBUF,X
1512  C7A5  10 F1              BPL GN37

1514  C7A7  20 B5 C4    GN40   JSR FNDFIL
1515  C7AA  38                 SEC
1516  C7AB  60          GN45   RTS

1518  C7AC  A0 1B       BLKNB  LDY #NBSIZ       ;BLANK NAMBUF
1519  C7AE  A9 20              LDA #$20
1520  C7B0  99 B0 02    BLKNB1 STA NAMBUF-1,Y
1521  C7B3  88                 DEY
1522  C7B4  D0 FA              BNE BLKNB1
1523  C7B6  60                 RTS

1525  C7B7              ;NEW DIRECTORY IN LISTING
1526  C7B7              NEWDIR
1527  C7B7  20 67 F1           JSR BAM2X
1528  C7BA  20 2D F1           JSR REDBAM
1529  C7BD  20 AC C7           JSR BLKNB
1530  C7C0  A9 FF              LDA #$FF
1531  C7C2  85 6F              STA TEMP
1532  C7C4  A6 7F              LDX DRVNUM
1533  C7C6  8E 72 02           STX NBTEMP
1534  C7C9  A9 00              LDA #0
1535  C7CB  8D 73 02           STA NBTEMP+1
1536  C7CE  A6 F9              LDX JOBNUM
1537  C7D0  BD 2F FF           LDA BUFIND,X
1538  C7D3  85 95              STA DIRBUF+1
1539  C7D5  AD D7 FE           LDA DSKNAM
1540  C7D8  85 94              STA DIRBUF
1541  C7DA  A0 16              LDY #22

1543  C7DC  B1 94       ND10   LDA (DIRBUF),Y
1544  C7DE  C9 A0              CMP #$A0
1545  C7E0  D0 0B              BNE ND20
1546  C7E2  A9 31              LDA #'1'        ;VERSION # 1
1547  C7E4  2C                 .BYTE $2C       ;SKIP NEXT INSTRUCTION
1548  C7E5              ND15
1549  C7E5  B1 94              LDA (DIRBUF),Y
1550  C7E7  C9 A0              CMP #$A0
1551  C7E9  D0 02              BNE ND20
1552  C7EB              ;
1553  C7EB  A9 20              LDA #$20
1554  C7ED              ND20
1555  C7ED  99 B3 02           STA NAMBUF+2,Y
1556  C7F0  88                 DEY
1557  C7F1  10 F2              BPL ND15
1558  C7F3  A9 12              LDA #$12
1559  C7F5  8D B1 02           STA NAMBUF
1560  C7F8  A9 22              LDA #'"'
1561  C7FA  8D B2 02           STA NAMBUF+1
```

```
LINE# LOC   CODE        LINE


1562  C7FD  8D C3 02           STA NAMBUF+18
1563  C800  A9 20              LDA #$20
1564  C802  8D C4 02           STA NAMBUF+19
1565  C805  60                 RTS

1567  C806  20 AC C7    MSGFRE JSR BLKNB
1568  C809  A0 0B              LDY #MSGLEN-1
1569  C80B  B9 17 C8    MSG1   LDA FREMSG,Y
1570  C80E  99 B1 02           STA NAMBUF,Y
1571  C811  88                 DEY
1572  C812  10 F7              BPL MSG1
1573  C814  4C 9B EF           JMP NUMFRE

1575  C817              ; TEXT: "BLOCKS FREE."
1576  C817  42 4C       FREMSG .BYTE 'BLOCKS FREE.'
1577  C823              MSGLEN =*-FREMSG
1578  C823              ;.END
1578  C823              ;
1579  C823                     .LIB SCRTCH
```

```
LINE# LOC   CODE         LINE


1581  C823               ; SCRATCH FILE(S)
1582  C823  20 98 C3     SCRTCH JSR FS1SET       ; SET UP FOR 1 STREAM
1583  C826  20 20 C3            JSR ALLDRS
1584  C829  20 CA C3            JSR OPTSCH
1585  C82C  A9 00               LDA #0
1586  C82E  85 86               STA R0          ; USED AS FILE COUNT
1587  C830  20 9D C4            JSR FFST
1588  C833  30 3D               BMI SC30
1589  C835  20 B7 DD     SC15   JSR TSTCHN       ; IS IT ACTIVE ?
1590  C838  90 33               BCC SC25        ; YES - DONT SCRATCH
1591  C83A  A0 00               LDY #0
1592  C83C  B1 94               LDA (DIRBUF),Y
1593  C83E  29 40               AND #$40        ; LOCK BIT
1594  C840  D0 2B               BNE SC25        ; IT'S LOCKED
1595  C842               ;
1596  C842  20 B6 C8            JSR DELDIR      ; DELETE DIRECTORY
1597  C845  A0 13               LDY #19         ; IS THIS A RELATIVE ?
1598  C847  B1 94               LDA (DIRBUF),Y ; HAS A SS ?
1599  C849  F0 0A               BEQ SC17        ; NO
1600  C84B  85 80               STA TRACK       ; YES - SAVE TRACK
1601  C84D  C8                  INY
1602  C84E  B1 94               LDA (DIRBUF),Y ; GET SECTOR
1603  C850  85 81               STA SECTOR
1604  C852  20 7D C8            JSR DELFIL      ; DELETE BY LINKS
1605  C855  AE 53 02     SC17   LDX ENTFND
1606  C858  A9 20               LDA #$20
1607  C85A  35 E7               AND PATTYP,X
1608  C85C  D0 0D               BNE SC20        ; CREATED, NOT CLOSED
1609  C85E               ;
1610  C85E  BD 80 02            LDA FILTRK,X    ; DELETE BY LINKS
1611  C861  85 80               STA TRACK
1612  C863  BD 85 02            LDA FILSEC,X
1613  C866  85 81               STA SECTOR
1614  C868  20 7D C8            JSR DELFIL

1616  C86B  E6 86        SC20   INC R0
1617  C86D  20 8B C4     SC25   JSR FFRE
1618  C870  10 C3               BPL SC15

1620  C872  A5 86        SC30   LDA R0           ; FINISHED, SET
1621  C874  85 80               STA TRACK       ; FILE COUNT
1622  C876  A9 01               LDA #1
1623  C878  A0 00               LDY #0
1624  C87A  4C A3 C1            JMP SCREND      ; END OF SCRATCH

1626  C87D  20 AD EF     DELFIL JSR FRETS        ; DELETE FILE BY LINKS
1627  C880  20 75 D4            JSR OPNIRD      ; UPDATE BAM
1628  C883  20 67 F1            JSR BAM2X
1629  C886  B5 A7               LDA BUF0,X
1630  C888  C9 FF               CMP #$FF
1631  C88A  F0 08               BEQ DEL10
1632  C88C  AD F9 02            LDA WBAM
1633  C88F  09 40               ORA #$40
1634  C891  8D F9 02            STA WBAM
1635  C894               DEL10
```

```
LINE# LOC   CODE        LINE


1636  C894  A9 00       DEL2   LDA #0
1637  C896  20 C8 D4           JSR SETPNT
1638  C899  20 56 D1           JSR RDBYT
1639  C89C  85 80              STA TRACK
1640  C89E  20 56 D1           JSR RDBYT
1641  C8A1  85 81              STA SECTOR
1642  C8A3  A5 80              LDA TRACK
1643  C8A5  D0 06              BNE DEL1
1644  C8A7  20 42 EF           JSR MAPOUT
1645  C8AA  4C 27 D2           JMP FRECHN
1646  C8AD  20 AD EF    DEL1   JSR FRETS
1647  C8B0  20 4D D4           JSR NXTBUF
1648  C8B3  4C 94 C8           JMP DEL2

1650  C8B6  A0 00       DELDIR LDY #0           ; DELETE DIR ENTRY
1651  C8B8  98                 TYA
1652  C8B9  91 94              STA (DIRBUF),Y
1653  C8BB  20 5E DE           JSR WRTOUT
1654  C8BE  4C 99 D5           JMP WATJOB

1656  C8C1              ;.END
1656  C8C1              ;
1657  C8C1                     .LIB DUPLCT
```

```
LINE# LOC   CODE        LINE


1659  C8C1              ; DUPLICATE DISK

1661  C8C1              DUPLCT
1662  C8C1  A9 31              LDA #BADCMD
1663  C8C3  4C C8 C1           JMP CMDERR

1665  C8C6              ; TRANSFER FORMAT CODE TO BUFFER 0
1666  C8C6              ; & START CONTROLLER FORMATTING
1667  C8C6              FORMAT
1668  C8C6  A9 4C              LDA #$4C
1669  C8C8  8D 00 06           STA BUFS+$300
1670  C8CB  A9 15              LDA #<FORMT
1671  C8CD  8D 01 06           STA BUFS+$301
1672  C8D0  A9 FB              LDA #>FORMT
1673  C8D2  8D 02 06           STA BUFS+$302
1674  C8D5              ;
1675  C8D5  A9 03              LDA #3
1676  C8D7  20 D3 D6           JSR SETH
1677  C8DA  A5 7F              LDA DRVNUM
1678  C8DC  09 E0              ORA #EXEC
1679  C8DE  85 03              STA JOBS+3
1680  C8E0  A5 03       FMT105 LDA JOBS+3
1681  C8E2  30 FC              BMI FMT105
1682  C8E4  C9 02              CMP #2
1683  C8E6  90 07              BCC FMT110
1684  C8E8  A9 03              LDA #3
1685  C8EA  A2 00              LDX #0
1686  C8EC  4C 0A E6           JMP ERROR
1687  C8EF  60          FMT110 RTS
1688  C8F0              ;
1689  C8F0              ;
1690  C8F0              ;.END
1690  C8F0              ;
1691  C8F0                     .LIB COPSET
```

```
LINE# LOC   CODE        LINE


1693  C8F0              ;
1694  C8F0              ;DSKCPY CHECK FOR TYPE
1695  C8F0              ;AND PARSES SPECIAL CASE
1696  C8F0              ;
1697  C8F0              DSKCPY
1698  C8F0  A9 E0              LDA #$E0       ;KILL BAM BUFFER
1699  C8F2  8D 4F 02           STA BUFUSE
1700  C8F5  20 1F F1           JSR CLNBAM     ;CLR TBAM
1701  C8F8  20 67 F1           JSR BAM2X      ;GET BAM LINDX IN .X
1702  C8FB  A9 FF              LDA #$FF
1703  C8FD  95 A7              STA BUF0,X     ;MARK BAM OUT-OF-MEMORY
1704  C8FF  A9 0F              LDA #$0F
1705  C901  8D 56 02           STA LINUSE     ;FREE ALL LINDXS
1706  C904  20 E5 C1           JSR PRSCLN     ;FIND ":"
1707  C907  D0 03              BNE DX0000
1708  C909  4C C1 C8           JMP DUPLCT     ;BAD COMMAND ERROR, CX=X NOT
                                               ALLOWED
1709  C90C              ;
1710  C90C              ;JSR PRSEQ
1711  C90C              ;
1712  C90C              ;LDA #'* ;CPY ALL
1713  C90C              ;LDX #39 ;PUT AT BUFFER END
1714  C90C              ;STX FILTBL+1
1715  C90C              ;STA CMDBUF,X ;PLACE *
1716  C90C              ;INX
1717  C90C              ;STX CMDSIZ
1718  C90C              ;LDX #1 ;SET UP CNT'S
1719  C90C              ;STX F1CNT
1720  C90C              ;INX
1721  C90C              ;STX F2CNT
1722  C90C              ;JMP MOVLP2 ;ENTER ROUTINE
1723  C90C              ;
1724  C90C  20 F8 C1    DX0000 JSR TC30        ;NORMAL PARSE
1725  C90F  20 20 C3    DX0005 JSR ALLDRS      ;PUT DRV'S IN FILTBL
1726  C912  AD 8B 02           LDA IMAGE      ;GET PARSE IMAGE
1727  C915  29 55              AND #%01010101 ;VAL FOR PATT COPY
1728  C917  D0 0F              BNE DX0020     ;MUST BE CONCAT OR NORMAL
1729  C919  AE 7A 02           LDX FILTBL     ;CHK FOR *
1730  C91C  BD 00 02           LDA CMDBUF,X    ;GET CHARACTER
1731  C91F  C9 2A              CMP #'*
1732  C921  D0 05              BNE DX0020
1733  C923  A9 30       DX0010 LDA #BADSYN     ;SYNTAX ERROR
1734  C925  4C C8 C1           JMP CMDERR
1735  C928  AD 8B 02    DX0020 LDA IMAGE       ;CHK FOR NORMAL
1736  C92B  29 D9              AND #%11011001
1737  C92D  D0 F4              BNE DX0010
1738  C92F  4C 52 C9           JMP COPY
1739  C932              ;.END
1740  C932              ;PRSEQ
1741  C932              ; LDA #'= ;SPECIAL CASE
1742  C932              ; JSR PARSE
1743  C932              ; BNE X0020
1744  C932              ;X0015 LDA #BADSYN
1745  C932              ; JMP CMDERR
1746  C932              ;X0020 LDA CMDBUF,Y
1747  C932              ; JSR TST0V1
```

```
LINE# LOC   CODE         LINE


1748  C932              ; BMI X0015
1749  C932              ; STA FILDRV+1 ;SRC DRV
1750  C932              ; DEY
1751  C932              ; DEY
1752  C932              ; LDA CMDBUF,Y
1753  C932              ; JSR TST0V1
1754  C932              ; BMI X0015
1755  C932              ; CMP FILDRV+1 ;CANNOT BE EQUAL
1756  C932              ; BEQ X0015
1757  C932              ; STA FILDRV ;DEST DRV
1758  C932              ; RTS
1759  C932              ;; .END
1759  C932              ;
1760  C932                     .LIB COPALL
```

```
LINE# LOC   CODE        LINE


1762  C932              ;
1763  C932              ; SET UP SUBROUTINE
1764  C932              ;
1765  C932  A9 00       PUPS1  LDA #0
1766  C934  8D 58 02           STA REC
1767  C937  8D 8C 02           STA DRVCNT
1768  C93A  8D 80 02           STA FILTRK
1769  C93D  8D 81 02           STA FILTRK+1
1770  C940  A5 E3              LDA FILDRV+1
1771  C942  29 01              AND #1
1772  C944  85 7F              STA DRVNUM
1773  C946  09 01              ORA #1
1774  C948  8D 91 02           STA DELSEC     ;NONZERO
1775  C94B  AD 7B 02           LDA FILTBL+1   ;FN1=FN2
1776  C94E  8D 7A 02           STA FILTBL
1777  C951  60                 RTS
1778  C952              ;
1779  C952              ;.END
1780  C952              ; .PAGE 'COPY ALL'
1781  C952              ;
1782  C952              ;COPY DISK TO DISK ROUTINES
1783  C952              ;
1784  C952              ;CPYDTD LDA FILTBL+1 ;SAVE IN TEMP
1785  C952              ; STA TEMP
1786  C952              ; LDY #40 ;40 CHAR BUFFER
1787  C952              ; LDX CMDSIZ ;PREP TO MOVE
1788  C952              ; STY CMDSIZ ;END OF FILENAME2
1789  C952              ;MOVLP1 DEY
1790  C952              ; DEX
1791  C952              ; LDA CMDBUF,X ;MOV FN LIFO
1792  C952              ; STA CMDBUF,Y
1793  C952              ; CPX TEMP ;ACTUAL F2 VAL
1794  C952              ; BNE MOVLP1
1795  C952              ; STY FILTBL+1 ;POINTER TO F2
1796  C952              ;MOVLP2 JSR OPTSCH
1797  C952              ; JSR PUPS1 ;SETUP FIRST PASS
1798  C952              ; JSR FFST ;FIRST MATCH
1799  C952              ; BPL FIXIT ;ENTRY FOUND?
1800  C952              ; BMI ENDIT ;NO
1801  C952              ;;
1802  C952              ;EXLP0 PLA ;PULL NEEDED VARS
1803  C952              ; STA DIRSEC
1804  C952              ; PLA
1805  C952              ; STA FILTBL+1
1806  C952              ; PLA
1807  C952              ; STA LSTBUF
1808  C952              ; PLA
1809  C952              ; STA FILCNT
1810  C952              ; PLA
1811  C952              ; STA INDEX
1812  C952              ; PLA
1813  C952              ; STA FOUND
1814  C952              ; PLA
1815  C952              ; STA DELIND
1816  C952              ; PLA
```

```
LINE# LOC   CODE         LINE


1817  C952               ; STA DRVFLG
1818  C952               ;;
1819  C952               ;EXLP1 JSR PUPS1 ;SET UP VARS
1820  C952               ; JSR FFRE ;NEXT MATCH
1821  C952               ; BPL FIXIT ;FOUND ONE?
1822  C952               ;ENDIT JMP ENDCMD ;NO! SO BYE
1823  C952               ;;
1824  C952               ;FIXIT LDA DRVFLG ;PUSH NEEDED VARS
1825  C952               ; PHA
1826  C952               ; LDA DELIND
1827  C952               ; PHA
1828  C952               ; LDA FOUND
1829  C952               ; PHA
1830  C952               ; LDA INDEX
1831  C952               ; PHA
1832  C952               ; LDA FILCNT
1833  C952               ; PHA
1834  C952               ; LDA LSTBUF
1835  C952               ; PHA
1836  C952               ; LDA FILTBL+1
1837  C952               ; PHA
1838  C952               ; LDA DIRSEC
1839  C952               ; PHA
1840  C952               ;;
1841  C952               ;EXLP2 JSR TRFNME ;TRANSFER NAME
1842  C952               ; LDA #1 ;FAKE OUT LOOKUP
1843  C952               ; STA F1CNT
1844  C952               ; STA F2CNT
1845  C952               ; JSR LOOKUP
1846  C952               ; LDA #1
1847  C952               ; STA F1CNT
1848  C952               ; LDA #2 ;REAL
1849  C952               ; STA F2CNT
1850  C952               ; JSR CY ;COPY IT
1851  C952               ; JMP EXLP0 ;NEXT ONE FOLKS
1852  C952               ;;
1853  C952               ;;TRANSFER NAME (DIRBUF) TO CMDBUF
1854  C952               ;;
1855  C952               ;TRFNME LDY #3 ;BOTH INDEXES
1856  C952               ; STY FILTBL ;BEGINING OF FILENAME1
1857  C952               ;TRF0 LDA (DIRBUF),Y ;MOVE IT
1858  C952               ; STA CMDBUF,Y
1859  C952               ; INY
1860  C952               ; CPY #19 ;ALL 16 CHARS PASSED?
1861  C952               ; BNE TRF0
1862  C952               ; RTS
1863  C952               ;;.END
1863  C952               ;
1864  C952                      .LIB RENAME
```

```
LINE# LOC   CODE        LINE


1866  C952              ; COPY FILE(S) TO ONE FILE
1867  C952              ;
1868  C952              COPY                       ;FILENAMES, OPTIMIZE
1869  C952  20 4F C4           JSR LOOKUP     ;LOOK UP ALL FILES
1870  C955  AD 78 02           LDA F2CNT
1871  C958  C9 03              CMP #3
1872  C95A  90 45              BCC COP10
1873  C95C              ;
1874  C95C  A5 E2              LDA FILDRV
1875  C95E  C5 E3              CMP FILDRV+1
1876  C960  D0 3F              BNE COP10
1877  C962              ;
1878  C962  A5 DD              LDA ENTIND
1879  C964  C5 DE              CMP ENTIND+1
1880  C966  D0 39              BNE COP10
1881  C968              ;
1882  C968  A5 D8              LDA ENTSEC
1883  C96A  C5 D9              CMP ENTSEC+1
1884  C96C  D0 33              BNE COP10
1885  C96E              ;
1886  C96E  20 CC CA           JSR CHKIN      ;CONCAT
1887  C971  A9 01              LDA #1
1888  C973  8D 79 02           STA F2PTR
1889  C976  20 FA C9           JSR OPIRFL
1890  C979              ;
1891  C979  20 25 D1           JSR TYPFIL
1892  C97C  F0 04              BEQ COP01
1893  C97E  C9 02              CMP #PRGTYP
1894  C980  D0 05              BNE COP05
1895  C982              COP01
1896  C982  A9 64              LDA #MISTYP
1897  C984  20 C8 C1           JSR CMDERR
1898  C987              COP05
1899  C987  A9 12              LDA #IWSA
1900  C989  85 83              STA SA
1901  C98B  AD 3C 02           LDA LINTAB+IRSA
1902  C98E  8D 3D 02           STA LINTAB+IWSA
1903  C991  A9 FF              LDA #$FF
1904  C993  8D 3C 02           STA LINTAB+IRSA
1905  C996  20 2A DA           JSR APPEND
1906  C999  A2 02              LDX #2
1907  C99B  20 B9 C9           JSR CY10
1908  C99E  4C 94 C1           JMP ENDCMD
1909  C9A1              COP10
1910  C9A1  20 A7 C9           JSR CY
1911  C9A4  4C 94 C1           JMP ENDCMD
1912  C9A7              ;
1913  C9A7              ;
1914  C9A7              CY
1915  C9A7  20 E7 CA           JSR CHKIO      ;CHECK FILES FOR EXISTENCE

1917  C9AA  A5 E2              LDA FILDRV
1918  C9AC  29 01              AND #1
1919  C9AE  85 7F              STA DRVNUM
1920  C9B0  20 86 D4           JSR OPNIWR     ; OPEN INTERNAL WRITE CHNL
```

```
LINE# LOC   CODE        LINE


1921  C9B3  20 E4 D6           JSR ADDFIL      ; ADD TO DIRECTORY
1922  C9B6  AE 77 02           LDX F1CNT

1924  C9B9  8E 79 02    CY10   STX F2PTR        ; SET UP READ FILE
1925  C9BC  20 FA C9           JSR OPIRFL
1926  C9BF              ;
1927  C9BF  A9 11              LDA #IRSA       ; ADD FOR REL COPY
1928  C9C1  85 83              STA SA
1929  C9C3  20 EB D0           JSR FNDRCH
1930  C9C6  20 25 D1           JSR TYPFIL
1931  C9C9  D0 03              BNE CY10A       ; NOT REL
1932  C9CB  20 53 CA           JSR CYEXT
1933  C9CE              ;
1934  C9CE  A9 08       CY10A  LDA #EOISND
1935  C9D0  85 F8              STA EOIFLG
1936  C9D2  4C D8 C9           JMP CY20
1937  C9D5              CY15
1938  C9D5  20 9B CF           JSR PIBYTE
1939  C9D8              CY20
1940  C9D8  20 35 CA           JSR GIBYTE
1941  C9DB  A9 80              LDA #LRF
1942  C9DD  20 A6 DD           JSR TSTFLG
1943  C9E0  F0 F3              BEQ CY15
1944  C9E2              ;
1945  C9E2  20 25 D1           JSR TYPFIL
1946  C9E5  F0 03              BEQ CY30
1947  C9E7              ;
1948  C9E7  20 9B CF           JSR PIBYTE
1949  C9EA              CY30
1950  C9EA  AE 79 02           LDX F2PTR
1951  C9ED              ;
1952  C9ED  E8                 INX
1953  C9EE  EC 78 02           CPX F2CNT
1954  C9F1  90 C6              BCC CY10        ; MORE FILES TO COPY
1955  C9F3  A9 12              LDA #IWSA
1956  C9F5  85 83              STA SA
1957  C9F7  4C 02 DB           JMP CLSCHN      ; CLOSE COPY CHANNEL, FILE
1958  C9FA              ;
1959  C9FA              OPIRFL
1960  C9FA  AE 79 02           LDX F2PTR
1961  C9FD  B5 E2              LDA FILDRV,X
1962  C9FF  29 01              AND #1
1963  CA01  85 7F              STA DRVNUM
1964  CA03  AD D4 FE           LDA DIRTRK
1965  CA06  85 80              STA TRACK
1966  CA08  B5 D8              LDA ENTSEC,X
1967  CA0A  85 81              STA SECTOR
1968  CA0C  20 75 D4           JSR OPNIRD
1969  CA0F  AE 79 02           LDX F2PTR
1970  CA12  B5 DD              LDA ENTIND,X
1971  CA14  20 C8 D4           JSR SETPNT
1972  CA17  AE 79 02           LDX F2PTR
1973  CA1A  B5 E7              LDA PATTYP,X
1974  CA1C  29 07              AND #TYPMSK
1975  CA1E  8D 4A 02           STA TYPE
```

```
LINE# LOC   CODE        LINE


1976  CA21              ;
1977  CA21  A9 00              LDA #0
1978  CA23  8D 58 02           STA REC
1979  CA26  20 A0 D9           JSR OPREAD
1980  CA29  A0 01              LDY #1
1981  CA2B  20 25 D1           JSR TYPFIL
1982  CA2E  F0 01              BEQ OPIR10
1983  CA30  C8                 INY
1984  CA31              OPIR10
1985  CA31  98                 TYA
1986  CA32  4C C8 D4           JMP SETPNT
1987  CA35              ;
1988  CA35              GIBYTE
1989  CA35  A9 11              LDA #IRSA
1990  CA37  85 83              STA SA
1991  CA39              GCBYTE
1992  CA39  20 9B D3           JSR GBYTE
1993  CA3C              ;
1994  CA3C  85 85              STA DATA
1995  CA3E  A6 82              LDX LINDX
1996  CA40  B5 F2              LDA CHNRDY,X
1997  CA42  29 08              AND #EOISND
1998  CA44  85 F8              STA EOIFLG
1999  CA46  D0 0A              BNE GIB20
2000  CA48              ;
2001  CA48  20 25 D1           JSR TYPFIL
2002  CA4B  F0 05              BEQ GIB20
2003  CA4D              ;
2004  CA4D  A9 80              LDA #LRF
2005  CA4F  20 97 DD           JSR SETFLG
2006  CA52  60          GIB20  RTS
2007  CA53              ;
2008  CA53  20 D3 D1    CYEXT  JSR SETDRN       ; COPY REL RECORDS
2009  CA56  20 CB E1           JSR SSEND
2010  CA59  A5 D6              LDA SSIND
2011  CA5B  48                 PHA
2012  CA5C  A5 D5              LDA SSNUM
2013  CA5E  48                 PHA
2014  CA5F  A9 12              LDA #IWSA
2015  CA61  85 83              STA SA
2016  CA63  20 07 D1           JSR FNDWCH
2017  CA66  20 D3 D1           JSR SETDRN
2018  CA69  20 CB E1           JSR SSEND
2019  CA6C  20 9C E2           JSR POSBUF
2020  CA6F  A5 D6              LDA SSIND
2021  CA71  85 87              STA R1
2022  CA73  A5 D5              LDA SSNUM
2023  CA75  85 86              STA R0
2024  CA77  A9 00              LDA #0
2025  CA79  85 88              STA R2
2026  CA7B  85 D4              STA RECPTR
2027  CA7D  85 D7              STA RELPTR
2028  CA7F  68                 PLA
2029  CA80  85 D5              STA SSNUM
2030  CA82  68                 PLA
```

```
LINE# LOC   CODE        LINE


2031  CA83  85 D6              STA SSIND
2032  CA85  4C 3B E3           JMP ADDR1
2033  CA88              ;
```

```
LINE# LOC   CODE        LINE


2035  CA88              ; RENAME FILE NAME IN DIRECTORY
2036  CA88  20 20 C3    RENAME JSR ALLDRS       ;SET BOTH DRIVE #'S
2037  CA8B  A5 E3              LDA FILDRV+1
2038  CA8D  29 01              AND #1
2039  CA8F  85 E3              STA FILDRV+1
2040  CA91  C5 E2              CMP FILDRV
2041  CA93  F0 02              BEQ RN10       ;SAME DRIVE #'S
2042  CA95  09 80              ORA #$80       ;CHECK BOTH DRIVES FOR NAME
2043  CA97  85 E2       RN10   STA FILDRV
2044  CA99  20 4F C4           JSR LOOKUP     ;LOOK UP BOTH NAMES
2045  CA9C  20 E7 CA           JSR CHKIO      ;CHECK FOR EXISTENCE
2046  CA9F  A5 E3              LDA FILDRV+1
2047  CAA1  29 01              AND #1
2048  CAA3  85 7F              STA DRVNUM
2049  CAA5  A5 D9              LDA ENTSEC+1
2050  CAA7  85 81              STA SECTOR
2051  CAA9  20 57 DE           JSR RDAB       ;READ DIRECTORY SECTOR
2052  CAAC  20 99 D5           JSR WATJOB
2053  CAAF  A5 DE              LDA ENTIND+1
2054  CAB1  18                 CLC            ; SET SECTOR INDEX
2055  CAB2  69 03              ADC #3         ; ...+3
2056  CAB4  20 C8 D4           JSR SETPNT
2057  CAB7  20 93 DF           JSR GETACT
2058  CABA  A8                 TAY
2059  CABB  AE 7A 02           LDX FILTBL
2060  CABE  A9 10              LDA #16
2061  CAC0  20 6E C6           JSR TRNAME     ;TRANSFER NAME
2062  CAC3  20 5E DE           JSR WRTOUT     ;WRITE SECTOR OUT
2063  CAC6  20 99 D5           JSR WATJOB
2064  CAC9  4C 94 C1           JMP ENDCMD


2067  CACC              ; CHECK I/O FILE FOR EXIST
2068  CACC              CHKIN
2069  CACC  A5 E8              LDA PATTYP+1   ;1ST FILE BEARS TYPE
2070  CACE  29 07              AND #TYPMSK
2071  CAD0  8D 4A 02           STA TYPE
2072  CAD3              ;
2073  CAD3  AE 78 02           LDX F2CNT
2074  CAD6  CA          CK10   DEX
2075  CAD7  EC 77 02           CPX F1CNT
2076  CADA  90 0A              BCC CK20
2077  CADC  BD 80 02           LDA FILTRK,X
2078  CADF  D0 F5              BNE CK10
2079  CAE1  A9 62              LDA #FLNTFD    ;INPUT FILE NOT FOUND
2080  CAE3  4C C8 C1           JMP CMDERR
2081  CAE6              CK20
2082  CAE6  60                 RTS
2083  CAE7              ;
2084  CAE7              CHKIO
2085  CAE7  20 CC CA           JSR CHKIN
2086  CAEA  BD 80 02    CK25   LDA FILTRK,X
2087  CAED  F0 05              BEQ CK30
2088  CAEF  A9 63              LDA #FLEXST
2089  CAF1  4C C8 C1           JMP CMDERR
```

```
LINE# LOC   CODE        LINE


2090  CAF4  CA          CK30   DEX
2091  CAF5  10 F3              BPL CK25
2092  CAF7  60                 RTS

2094  CAF8              ;.END
2094  CAF8              ;
2095  CAF8                     .LIB MEMRW
```

```
LINE# LOC   CODE         LINE


2097  CAF8               ; MEMORY ACCESS COMMANDS
2098  CAF8               ; "-" MUST BE 2ND CHAR
2099  CAF8  AD 01 02     MEM    LDA CMDBUF+1
2100  CAFB  C9 2D               CMP #'-'
2101  CAFD  D0 4C               BNE MEMERR
2102  CAFF               ;
2103  CAFF  AD 03 02            LDA CMDBUF+3   ; SET ADDRESS IN TEMP
2104  CB02  85 6F               STA TEMP
2105  CB04  AD 04 02            LDA CMDBUF+4
2106  CB07  85 70               STA TEMP+1
2107  CB09               ;
2108  CB09  A0 00               LDY #0
2109  CB0B  AD 02 02            LDA CMDBUF+2
2110  CB0E  C9 52               CMP #'R'
2111  CB10  F0 0E               BEQ MEMRD      ; READ
2112  CB12  20 A6 F2            JSR KILLP      ; KILL PROTECT
2113  CB15  C9 57               CMP #'W'
2114  CB17  F0 37               BEQ MEMWRT     ; WRITE
2115  CB19  C9 45               CMP #'E'
2116  CB1B  D0 2E               BNE MEMERR     ; ERROR
2117  CB1D               ; EXECUTE
2118  CB1D  6C 6F 00     MEMEX  JMP (TEMP)

2120  CB20               ; M-R COMMAND
2121  CB20               MEMRD
2122  CB20  B1 6F               LDA (TEMP),Y
2123  CB22  85 85               STA DATA
2124  CB24  AD 74 02            LDA CMDSIZ
2125  CB27  C9 06               CMP #6
2126  CB29  90 1A               BCC M30
2127  CB2B               ;
2128  CB2B  AE 05 02            LDX CMDBUF+5
2129  CB2E  CA                  DEX
2130  CB2F  F0 14               BEQ M30
2131  CB31  8A                  TXA
2132  CB32  18                  CLC
2133  CB33  65 6F               ADC TEMP
2134  CB35  E6 6F               INC TEMP
2135  CB37  8D 49 02            STA LSTCHR+ERRCHN
2136  CB3A  A5 6F               LDA TEMP
2137  CB3C  85 A5               STA CB+2
2138  CB3E  A5 70               LDA TEMP+1
2139  CB40  85 A6               STA CB+3
2140  CB42  4C 43 D4            JMP GE20
2141  CB45               M30
2142  CB45  20 EB D0            JSR FNDRCH
2143  CB48  4C 3A D4            JMP GE15

2145  CB4B  A9 31        MEMERR LDA #BADCMD     ; BAD COMMAND
2146  CB4D  4C C8 C1            JMP CMDERR

2148  CB50               ; M-W COMMAND
2149  CB50               MEMWRT                 ; WRITE
2150  CB50  B9 06 02     M10    LDA CMDBUF+6,Y
2151  CB53  91 6F               STA (TEMP),Y   ; TRANSFER FROM CMDBUF
```

```
LINE# LOC   CODE         LINE


2152  CB55  C8                  INY
2153  CB56  CC 05 02            CPY CMDBUF+5   ; # OF BYTES TO WRITE
2154  CB59  90 F5               BCC M10
2155  CB5B  60                  RTS

2157  CB5C               ;.END
2157  CB5C               ;
2158  CB5C                      .LIB BLOCK
```

```
LINE# LOC   CODE        LINE


2160  CB5C              ; ROM 1.1 ADDITIONS
2161  CB5C              ; USER COMMANDS
2162  CB5C              ;
2163  CB5C  AC 01 02    USER   LDY CMDBUF+1
2164  CB5F  C0 30              CPY #'0'
2165  CB61  D0 09              BNE US10       ; 0 RESETS PNTR
2166  CB63              ;
2167  CB63  A9 EA       USRINT LDA #<UBLOCK    ; SET DEFAULT BLOCK ADD
2168  CB65  85 6B              STA USRJMP
2169  CB67  A9 FF              LDA #>UBLOCK
2170  CB69  85 6C              STA USRJMP+1
2171  CB6B  60                 RTS
2172  CB6C                                    ;
2173  CB6C  20 72 CB    US10   JSR USREXC      ;EXECUTE CODE BY TABLE
2174  CB6F  4C 94 C1           JMP ENDCMD
2175  CB72                                    ;
2176  CB72  88          USREXC DEY             ;ENTRY IS (((INDEX-1)AND$F)X2)
2177  CB73  98                 TYA
2178  CB74  29 0F              AND #$F
2179  CB76  0A                 ASL A
2180  CB77  A8                 TAY
2181  CB78  B1 6B              LDA (USRJMP),Y
2182  CB7A  85 75              STA IP
2183  CB7C  C8                 INY
2184  CB7D  B1 6B              LDA (USRJMP),Y
2185  CB7F  85 76              STA IP+1
2186  CB81  6C 75 00           JMP (IP)
```

```
LINE# LOC   CODE        LINE


2188  CB84              ; OPEN DIRECT ACCESS BUFFER
2189  CB84              ; FROM OPEN "#"
2190  CB84              ;
2191  CB84  AD 8E 02    OPNBLK LDA LSTDRV
2192  CB87  85 7F              STA DRVNUM
2193  CB89  A5 83              LDA SA         ; SA IS DESTROYED BY THIS PATCH
2194  CB8B  48                 PHA
2195  CB8C  20 3D C6           JSR AUTOI      ; INIT DISK FOR PROPER CHANNEL
                                               ASSIGNMENT
2196  CB8F  68                 PLA            ; RESTORE SA
2197  CB90  85 83              STA SA
2198  CB92  AE 74 02           LDX CMDSIZ
2199  CB95  CA                 DEX
2200  CB96  D0 0D              BNE OB10
2201  CB98                  ;
2202  CB98  A9 01              LDA #1         ;GET ANY BUFFER
2203  CB9A  20 E2 D1           JSR GETRCH
2204  CB9D  4C F1 CB           JMP OB30
2205  CBA0                  ;
2206  CBA0  A9 70       OB05   LDA #NOCHNL
2207  CBA2  4C C8 C1           JMP CMDERR
2208  CBA5                  ;
2209  CBA5  A0 01       OB10   LDY #1          ; BUFFER # IS REQUESTED
2210  CBA7  20 7C CC           JSR BP05
2211  CBAA  AE 85 02           LDX FILSEC
2212  CBAD  E0 05              CPX #BFCNT     ; MUST BE LESS THAN 13.
2213  CBAF  B0 EF              BCS OB05
2214  CBB1                  ;
2215  CBB1  A9 00              LDA #0
2216  CBB3  85 6F              STA TEMP
2217  CBB5  85 70              STA TEMP+1
2218  CBB7  38                 SEC
2219  CBB8                  ;
2220  CBB8              OB15
2221  CBB8  26 6F              ROL TEMP
2222  CBBA  26 70              ROL TEMP+1
2223  CBBC  CA                 DEX
2224  CBBD  10 F9              BPL OB15
2225  CBBF                  ;
2226  CBBF  A5 6F              LDA TEMP
2227  CBC1  2D 4F 02           AND BUFUSE
2228  CBC4  D0 DA              BNE OB05       ; BUFFER IS USED
2229  CBC6  A5 70              LDA TEMP+1
2230  CBC8  2D 50 02           AND BUFUSE+1
2231  CBCB  D0 D3              BNE OB05       ; BUF IS USED
2232  CBCD                  ;
2233  CBCD  A5 6F              LDA TEMP
2234  CBCF  0D 4F 02           ORA BUFUSE     ; SET BUFFER AS USED
2235  CBD2  8D 4F 02           STA BUFUSE
2236  CBD5  A5 70              LDA TEMP+1
2237  CBD7  0D 50 02           ORA BUFUSE+1
2238  CBDA  8D 50 02           STA BUFUSE+1
2239  CBDD                  ;
2240  CBDD  A9 00              LDA #0         ; SET UP CHANNEL
2241  CBDF  20 E2 D1           JSR GETRCH
2242  CBE2  A6 82              LDX LINDX
```

```
LINE# LOC   CODE        LINE


2243  CBE4  AD 85 02           LDA FILSEC
2244  CBE7  95 A7              STA BUF0,X
2245  CBE9  AA                 TAX
2246  CBEA  A5 7F              LDA DRVNUM
2247  CBEC  95 00              STA JOBS,X
2248  CBEE  9D 5B 02           STA LSTJOB,X
2249  CBF1              ;
2250  CBF1  A6 83       OB30   LDX SA
2251  CBF3  BD 2B 02           LDA LINTAB,X   ; SET LINDX TABLE
2252  CBF6  09 40              ORA #$40
2253  CBF8  9D 2B 02           STA LINTAB,X
2254  CBFB              ;
2255  CBFB  A4 82              LDY LINDX
2256  CBFD  A9 FF              LDA #$FF
2257  CBFF  99 44 02           STA LSTCHR,Y
2258  CC02              ;
2259  CC02  A9 89              LDA #RNDRDY
2260  CC04  99 F2 00           STA CHNRDY,Y   ; SET CHANNEL READY
2261  CC07              ;
2262  CC07  B9 A7 00           LDA BUF0,Y
2263  CC0A  99 3E 02           STA CHNDAT,Y   ; BUFFER # AS 1ST CHAR
2264  CC0D  0A                 ASL A
2265  CC0E  AA                 TAX
2266  CC0F  A9 01              LDA #1
2267  CC11  95 99              STA BUFTAB,X
2268  CC13  A9 0E              LDA #DIRTYP+DIRTYP
2269  CC15  99 EC 00           STA FILTYP,Y   ; SET DIRECT FILE TYPE
2270  CC18  4C 94 C1           JMP ENDCMD
```

```
LINE# LOC   CODE        LINE


2272  CC1B              ;
2273  CC1B              ; BLOCK COMMANDS
2274  CC1B  A0 00       BLOCK  LDY #0
2275  CC1D  A2 00              LDX #0
2276  CC1F  A9 2D              LDA #'-         ; "-" SEPARATES CMD FROM SUBCMD
2277  CC21  20 68 C2           JSR PARSE       ; LOCATE SUB-CMD
2278  CC24  D0 0A              BNE BLK40
2279  CC26              ;
2280  CC26  A9 31       BLK10  LDA #BADCMD
2281  CC28  4C C8 C1           JMP CMDERR
2282  CC2B              ;
2283  CC2B  A9 30       BLK30  LDA #BADSYN
2284  CC2D  4C C8 C1           JMP CMDERR
2285  CC30                                     ;
2286  CC30  8A          BLK40  TXA
2287  CC31  D0 F8              BNE BLK30
2288  CC33                                     ;
2289  CC33  A2 05              LDX #NBCMDS-1   ; FIND COMMAND
2290  CC35  B9 00 02           LDA CMDBUF,Y
2291  CC38  DD 5D CC    BLK50  CMP BCTAB,X
2292  CC3B  F0 05              BEQ BLK60
2293  CC3D  CA                 DEX
2294  CC3E  10 F8              BPL BLK50
2295  CC40  30 E4              BMI BLK10
2296  CC42              ;
2297  CC42              BLK60
2298  CC42  8A                 TXA
2299  CC43  09 80              ORA #$80
2300  CC45  8D 2A 02           STA CMDNUM
2301  CC48  20 6F CC           JSR BLKPAR      ; PARSE PARMS
2302  CC4B              ;
2303  CC4B  AD 2A 02           LDA CMDNUM
2304  CC4E  0A                 ASL A
2305  CC4F  AA                 TAX
2306  CC50  BD 64 CC           LDA BCJMP+1,X
2307  CC53  85 70              STA TEMP+1
2308  CC55  BD 63 CC           LDA BCJMP,X
2309  CC58  85 6F              STA TEMP
2310  CC5A              ;
2311  CC5A  6C 6F 00           JMP (TEMP)      ; GOTO COMMAND
2312  CC5D              ;
2313  CC5D              ; TABLE OF POSSIBLE B- COMMANDS
2314  CC5D  41          BCTAB  .BYTE 'A'
2315  CC5E  46                 .BYTE 'F'
2316  CC5F  52                 .BYTE 'R'
2317  CC60  57                 .BYTE 'W'
2318  CC61  45                 .BYTE 'E'
2319  CC62  50                 .BYTE 'P'
2320  CC63              NBCMDS =*-BCTAB
2321  CC63              ;
2322  CC63  03 CD       BCJMP  .WORD BLKALC     ; BLOCK-ALLOCATE
2323  CC65  F5 CC              .WORD BLKFRE    ; BLOCK-FREE
2324  CC67  56 CD              .WORD BLKRD     ; BLOCK-READ
2325  CC69  73 CD              .WORD BLKWT     ; BLOCK-WRITE
2326  CC6B  A3 CD              .WORD BLKEXC    ; BLOCK-EXECUTE
```

```
LINE# LOC   CODE        LINE


2327  CC6D  BD CD              .WORD BLKPTR   ; BLOCK-POINTER
2328  CC6F              ;
2329  CC6F  A0 00       BLKPAR LDY #0          ; PARSE BLOCK PARMS
2330  CC71  A2 00              LDX #0
2331  CC73  A9 3A              LDA #':'
2332  CC75  20 68 C2           JSR PARSE
2333  CC78  D0 02              BNE BP05       ; FOUND ":"
2334  CC7A              ;
2335  CC7A  A0 03              LDY #3         ; ELSE CHAR #3 IS BEGINNING
2336  CC7C  B9 00 02    BP05   LDA CMDBUF,Y
2337  CC7F  C9 20              CMP #' '
2338  CC81  F0 08              BEQ BP10
2339  CC83              ;
2340  CC83  C9 1D              CMP #29        ; SKIP CHARACTER
2341  CC85  F0 04              BEQ BP10
2342  CC87              ;
2343  CC87  C9 2C              CMP #','
2344  CC89  D0 07              BNE BP20
2345  CC8B              ;
2346  CC8B  C8          BP10   INY
2347  CC8C  CC 74 02           CPY CMDSIZ
2348  CC8F  90 EB              BCC BP05
2349  CC91  60                 RTS            ;THAT'S ALL
2350  CC92              ;
2351  CC92  20 A1 CC    BP20   JSR ASCHEX
2352  CC95              ;
2353  CC95  EE 77 02           INC F1CNT
2354  CC98  AC 79 02           LDY F2PTR
2355  CC9B  E0 04              CPX #MXFILS-1
2356  CC9D  90 EC              BCC BP10
2357  CC9F              ;
2358  CC9F  B0 8A              BCS BLK30      ; BAD SYNTAX
2359  CCA1              ;
2360  CCA1              ; CONVERT ASCII TO HEX (BINARY)
2361  CCA1              ;  & STORE CONVERSION IN TABLES
2362  CCA1              ;  .Y= PTR INTO CMDBUF
2363  CCA1  A9 00       ASCHEX LDA #0
2364  CCA3  85 6F              STA TEMP
2365  CCA5  85 70              STA TEMP+1
2366  CCA7  85 72              STA TEMP+3
2367  CCA9              ;
2368  CCA9  A2 FF              LDX #$FF
2369  CCAB  B9 00 02    AH10   LDA CMDBUF,Y    ; TEST FOR DEC #
2370  CCAE  C9 40              CMP #$40
2371  CCB0  B0 18              BCS AH20       ; NON-NUMERIC TERMINATES
2372  CCB2  C9 30              CMP #$30
2373  CCB4  90 14              BCC AH20       ; NON-NUMERIC
2374  CCB6              ;
2375  CCB6  29 0F              AND #$F
2376  CCB8  48                 PHA
2377  CCB9  A5 70              LDA TEMP+1     ; SHIFT DIGITS (*10)
2378  CCBB  85 71              STA TEMP+2
2379  CCBD  A5 6F              LDA TEMP
2380  CCBF  85 70              STA TEMP+1
2381  CCC1  68                 PLA
```

```
LINE# LOC   CODE        LINE


2382  CCC2  85 6F              STA TEMP
2383  CCC4  C8                 INY
2384  CCC5  CC 74 02           CPY CMDSIZ
2385  CCC8  90 E1              BCC AH10       ; STILL IN STRING
2386  CCCA              ;
2387  CCCA  8C 79 02    AH20   STY F2PTR       ; CONVERT DIGIT TO...
2388  CCCD  18                 CLC            ; ...BINARY BY DEC TABLE
2389  CCCE  A9 00              LDA #0
2390  CCD0              ;
2391  CCD0  E8          AH30   INX
2392  CCD1  E0 03              CPX #3
2393  CCD3  B0 0F              BCS AH40
2394  CCD5              ;
2395  CCD5  B4 6F              LDY TEMP,X
2396  CCD7  88          AH35   DEY
2397  CCD8  30 F6              BMI AH30
2398  CCDA              ;
2399  CCDA  7D F2 CC           ADC DECTAB,X
2400  CCDD  90 F8              BCC AH35
2401  CCDF              ;
2402  CCDF  18                 CLC
2403  CCE0  E6 72              INC TEMP+3
2404  CCE2  D0 F3              BNE AH35
2405  CCE4              ;
2406  CCE4  48          AH40   PHA
2407  CCE5  AE 77 02           LDX F1CNT
2408  CCE8  A5 72              LDA TEMP+3
2409  CCEA  9D 80 02           STA FILTRK,X   ; STORE RESULT IN TABLE
2410  CCED  68                 PLA
2411  CCEE  9D 85 02           STA FILSEC,X
2412  CCF1  60                 RTS
2413  CCF2              ;
2414  CCF2              ; CONSTANTS FOR DEC TO HEX CONVERSION
2415  CCF2  01          DECTAB .BYTE 1
2416  CCF3  0A                 .BYTE 10
2417  CCF4  64                 .BYTE 100
2418  CCF5              ;
2419  CCF5              ;BLOCK-FREE
2420  CCF5  20 F5 CD    BLKFRE JSR BLKTST
2421  CCF8  20 AD EF           JSR FRETS
2422  CCFB  4C 94 C1           JMP ENDCMD
2423  CCFE              ;
2424  CCFE              ;BLOCK-ALLOCATE
2425  CCFE  A9 01              LDA #1
2426  CD00  8D F9 02           STA WBAM
2427  CD03              BLKALC
2428  CD03  20 F5 CD           JSR BLKTST
2429  CD06              ;
2430  CD06              BA10
2431  CD06  A5 81              LDA SECTOR
2432  CD08  48                 PHA
2433  CD09  20 48 F2           JSR GETSEC
2434  CD0C  F0 0B              BEQ BA15       ; NONE GREATER ON THIS TRACK
2435  CD0E  68                 PLA
2436  CD0F  C5 81              CMP SECTOR
```

```
LINE# LOC   CODE        LINE


2437  CD11  D0 19              BNE BA30       ; REQUESTED SECTOR NOT AVAIL
2438  CD13  20 DE EF           JSR WUSED
2439  CD16  4C 94 C1           JMP ENDCMD
2440  CD19              ;
2441  CD19              BA15
2442  CD19  68                 PLA            ; POP STACK
2443  CD1A              BA20
2444  CD1A  A9 00              LDA #0
2445  CD1C  85 81              STA SECTOR
2446  CD1E  E6 80              INC TRACK
2447  CD20  A5 80              LDA TRACK
2448  CD22  CD 26 FF           CMP MAXTRK
2449  CD25  B0 0A              BCS BA40       ; GONE ALL THE WAY
2450  CD27              ;
2451  CD27  20 48 F2           JSR GETSEC
2452  CD2A  F0 EE              BEQ BA20
2453  CD2C              BA30
2454  CD2C  A9 65              LDA #NOBLK
2455  CD2E  20 45 E6           JSR CMDER2
2456  CD31              BA40
2457  CD31  A9 65              LDA #NOBLK
2458  CD33  20 C8 C1           JSR CMDERR     ; T=0,S=0 :NONE LEFT
2459  CD36              ;
2460  CD36              ;
2461  CD36              ; BLOCK READ SUBS
2462  CD36  20 F2 CD    BLKRD2 JSR BKOTST      ; TEST PARMS
2463  CD39  4C 60 D4           JMP DRTRD
2464  CD3C              ;
2465  CD3C  20 2F D1    GETSIM JSR GETPRE      ; GET BYTE W/O INC
2466  CD3F  A1 99              LDA (BUFTAB,X)
2467  CD41  60                 RTS
2468  CD42              ;
2469  CD42              ; BLOCK READ
2470  CD42  20 36 CD    BLKRD3 JSR BLKRD2
2471  CD45  A9 00              LDA #0
2472  CD47  20 C8 D4           JSR SETPNT
2473  CD4A  20 3C CD           JSR GETSIM     ; Y=LINDX
2474  CD4D              ;
2475  CD4D              ;
2476  CD4D  99 44 02           STA LSTCHR,Y
2477  CD50  A9 89              LDA #RNDRDY
2478  CD52  99 F2 00           STA CHNRDY,Y
2479  CD55  60                 RTS
2480  CD56              ;
2481  CD56              ; BLOCK-READ
2482  CD56              BLKRD
2483  CD56  20 42 CD           JSR BLKRD3
2484  CD59  20 EC D3           JSR RNGET1
2485  CD5C  4C 94 C1           JMP ENDCMD
2486  CD5F              ;
2487  CD5F              ;USER DIRECT READ, LSTCHR=$FF
2488  CD5F              UBLKRD
2489  CD5F  20 6F CC           JSR BLKPAR
2490  CD62  20 42 CD           JSR BLKRD3
2491  CD65  B9 44 02           LDA LSTCHR,Y
```

```
LINE# LOC   CODE        LINE


2492  CD68  99 3E 02           STA CHNDAT,Y
2493  CD6B  A9 FF              LDA #$FF
2494  CD6D  99 44 02           STA LSTCHR,Y
2495  CD70  4C 94 C1           JMP ENDCMD     ; (RTS)
2496  CD73              ;
2497  CD73              ;BLOCK-WRITE
2498  CD73  20 F2 CD    BLKWT  JSR BKOTST
2499  CD76              ;
2500  CD76  20 E8 D4           JSR GETPNT
2501  CD79  A8                 TAY
2502  CD7A  88                 DEY
2503  CD7B  C9 02              CMP #2
2504  CD7D  B0 02              BCS BW10
2505  CD7F  A0 01              LDY #1
2506  CD81              ;
2507  CD81  A9 00       BW10   LDA #0           ; SET RECORD SIZE
2508  CD83  20 C8 D4           JSR SETPNT
2509  CD86  98                 TYA
2510  CD87  20 F1 CF           JSR PUTBYT
2511  CD8A  8A                 TXA
2512  CD8B  48                 PHA
2513  CD8C              ;
2514  CD8C  20 64 D4    BW20   JSR DRTWRT      ; WRITE BLOCK
2515  CD8F  68                 PLA
2516  CD90  AA                 TAX
2517  CD91  20 EE D3           JSR RNGET2
2518  CD94  4C 94 C1           JMP ENDCMD
2519  CD97              ;
2520  CD97              ;USER DIRCT WRITE, NO LSTCHR
2521  CD97  20 6F CC    UBLKWT JSR BLKPAR
2522  CD9A  20 F2 CD           JSR BKOTST
2523  CD9D  20 64 D4           JSR DRTWRT
2524  CDA0  4C 94 C1           JMP ENDCMD
2525  CDA3              ;
2526  CDA3              ;IN .FILE VECTOR:
2527  CDA3              ;*=$FFFA-6 ;USER DIRECT ACCESS
2528  CDA3              ;UBLOCK .WORD UBLKRD
2529  CDA3              ;       .WORD UBLKWT
2530  CDA3              ;
2531  CDA3              ;BLOCK-EXECUTE
2532  CDA3              BLKEXC
2533  CDA3  20 A6 F2           JSR KILLP      ; KILL PROTECT
2534  CDA6  20 36 CD           JSR BLKRD2     ; READ BLOCK & EXECUTE
2535  CDA9  A9 00              LDA #0
2536  CDAB              ;
2537  CDAB  85 6F       BE05   STA TEMP
2538  CDAD  A6 F9              LDX JOBNUM
2539  CDAF  BD 2F FF           LDA BUFIND,X
2540  CDB2  85 70              STA TEMP+1
2541  CDB4  20 BA CD           JSR BE10       ; INDIRECT JSR
2542  CDB7  4C 94 C1           JMP ENDCMD
2543  CDBA              ;
2544  CDBA  6C 6F 00    BE10   JMP (TEMP)
2545  CDBD              ;
2546  CDBD              ;BUFFER-POINTER, SET BUFFER POINTER
```

```
LINE# LOC   CODE        LINE


2547  CDBD  20 D2 CD    BLKPTR JSR BUFTST
2548  CDC0  A5 F9              LDA JOBNUM
2549  CDC2  0A                 ASL A
2550  CDC3  AA                 TAX
2551  CDC4  AD 86 02           LDA FILSEC+1
2552  CDC7  95 99              STA BUFTAB,X
2553  CDC9  20 2F D1           JSR GETPRE
2554  CDCC  20 EE D3           JSR RNGET2     ; SET UP GET
2555  CDCF  4C 94 C1           JMP ENDCMD
2556  CDD2              ;
2557  CDD2              ;TEST FOR ALLOCATED BUFFER..
2558  CDD2              ; ..RELATED TO SA
2559  CDD2  A6 D3       BUFTST LDX F1PTR
2560  CDD4  E6 D3              INC F1PTR
2561  CDD6  BD 85 02           LDA FILSEC,X
2562  CDD9  A8                 TAY
2563  CDDA  88                 DEY
2564  CDDB  88                 DEY
2565  CDDC  C0 0C              CPY #$C        ; SET LIMIT TO # OF SAS
2566  CDDE  90 05              BCC BT20
2567  CDE0              ;
2568  CDE0  A9 70       BT15   LDA #NOCHNL
2569  CDE2  4C C8 C1           JMP CMDERR
2570  CDE5              ;
2571  CDE5  85 83       BT20   STA SA
2572  CDE7  20 EB D0           JSR FNDRCH
2573  CDEA  B0 F4              BCS BT15
2574  CDEC  20 93 DF           JSR GETACT
2575  CDEF  85 F9              STA JOBNUM
2576  CDF1  60                 RTS
2577  CDF2              ;
2578  CDF2              ;TEST BLOCK OPERATION PARMS
2579  CDF2  20 D2 CD    BKOTST JSR BUFTST
2580  CDF5              ;
2581  CDF5              ;TEST FOR LEGAL BLOCK &..
2582  CDF5              ; ..SET UP DRV, TRK, SEC
2583  CDF5  A6 D3       BLKTST LDX F1PTR
2584  CDF7  BD 85 02           LDA FILSEC,X
2585  CDFA  29 01              AND #1
2586  CDFC  85 7F              STA DRVNUM
2587  CDFE  BD 87 02           LDA FILSEC+2,X
2588  CE01  85 81              STA SECTOR
2589  CE03  BD 86 02           LDA FILSEC+1,X
2590  CE06  85 80              STA TRACK
2591  CE08              BT05
2592  CE08  20 5F D5           JSR TSCHK
2593  CE0B  4C 00 C1           JMP SETLDS     ; (RTS)
2594  CE0E              ;
2595  CE0E              ;      .END
2596  CE0E              ; RSR 1/19/80 ADD AUTOI TO #CMD
2596  CE0E              ;
2597  CE0E                     .LIB FNDREL
```

```
LINE# LOC   CODE         LINE


2599  CE0E               ;********************************
2600  CE0E               ;*
2601  CE0E               ;* FIND RELATIVE FILE
2602  CE0E               ;*
2603  CE0E               ;* VERSION 2.5
2604  CE0E               ;*
2605  CE0E               ;*
2606  CE0E               ;* INPUTS
2607  CE0E               ;* RECL - 1BYTE=LO RECORD #
2608  CE0E               ;* RECH - 1BYTE=HI RECORD #
2609  CE0E               ;* RS - 1BYTE=RECORD SIZE
2610  CE0E               ;* RECPTR - 1BYTE=FIRST BYTE
2611  CE0E               ;* WANTED FROM RECORD
2612  CE0E               ;*
2613  CE0E               ;* OUTPUTS
2614  CE0E               ;* SSNUM - 1BYTE=SIDE SECTOR #
2615  CE0E               ;* SSIND - 1BYTE=INDEX INTO SS
2616  CE0E               ;* RELPTR - 1BYTE=PTR TO FIRST
2617  CE0E               ;* BYTE WANTED
2618  CE0E               ;*
2619  CE0E               ;********************************
2620  CE0E               ;
2621  CE0E               ;
2622  CE0E               ;
2623  CE0E               ;
2624  CE0E               ;
2625  CE0E               ;
2626  CE0E  20 2C CE     FNDREL JSR MULPLY      ;RESULT=RN*RS+RP
2627  CE11  20 6E CE            JSR DIV254     ;DIVIDE BY 254
2628  CE14  A5 90               LDA ACCUM+1    ;SAVE REMAINDER
2629  CE16  85 D7               STA RELPTR
2630  CE18  20 71 CE            JSR DIV120     ;DIVIDE BY 120
2631  CE1B  E6 D7               INC RELPTR
2632  CE1D  E6 D7               INC RELPTR
2633  CE1F  A5 8B               LDA RESULT     ;SAVE QUOTIENT
2634  CE21  85 D5               STA SSNUM
2635  CE23  A5 90               LDA ACCUM+1    ;SAVE REMAINDER
2636  CE25  0A                  ASL A          ;CALC INDEX INTO SS
2637  CE26  18                  CLC
2638  CE27  69 10               ADC #16        ;SKIP LINK TABLE
2639  CE29  85 D6               STA SSIND
2640  CE2B  60                  RTS
2641  CE2C               ;
2642  CE2C               ;
2643  CE2C               ;
2644  CE2C               ;MULTIPLY
2645  CE2C               ;
2646  CE2C               ;RESULT=RECNUM*RS+RECPTR
2647  CE2C               ;
2648  CE2C               ; DESTROYS A,X
2649  CE2C               ;
2650  CE2C  20 D9 CE     MULPLY JSR ZERRES      ;RESULT=0
2651  CE2F  85 92               STA ACCUM+3    ;A=0
2652  CE31  A6 82               LDX LINDX      ;GET INDEX
2653  CE33  B5 B5               LDA RECL,X
```

```
LINE# LOC   CODE         LINE


2654  CE35  85 90               STA ACCUM+1
2655  CE37  B5 BB               LDA RECH,X
2656  CE39  85 91               STA ACCUM+2
2657  CE3B  D0 04               BNE MUL25      ;ADJUST FOR REC #1 &...
2658  CE3D  A5 90               LDA ACCUM+1    ;...#0 = 1ST REC
2659  CE3F  F0 0B               BEQ MUL50
2660  CE41  A5 90        MUL25  LDA ACCUM+1
2661  CE43  38                  SEC
2662  CE44  E9 01               SBC #1
2663  CE46  85 90               STA ACCUM+1
2664  CE48  B0 02               BCS MUL50
2665  CE4A  C6 91               DEC ACCUM+2
2666  CE4C               MUL50
2667  CE4C  B5 C7               LDA RS,X       ;COPY RECSIZ
2668  CE4E  85 6F               STA TEMP
2669  CE50  46 6F        MUL100 LSR TEMP        ;DO AN ADD ?
2670  CE52  90 03               BCC MUL200     ;NO
2671  CE54  20 ED CE            JSR ADDRES     ;RESULT=RESULT+ACCUM+1,2,3
2672  CE57  20 E5 CE     MUL200 JSR ACCX2       ;2*(ACCUM+1,2,3)
2673  CE5A  A5 6F               LDA TEMP       ;DONE ?
2674  CE5C  D0 F2               BNE MUL100     ;NO
2675  CE5E  A5 D4               LDA RECPTR     ;ADD IN LAST BIT
2676  CE60  18                  CLC
2677  CE61  65 8B               ADC RESULT
2678  CE63  85 8B               STA RESULT
2679  CE65  90 06               BCC MUL400     ;SKIP NO CARRY
2680  CE67  E6 8C               INC RESULT+1
2681  CE69  D0 02               BNE MUL400
2682  CE6B  E6 8D               INC RESULT+2
2683  CE6D  60           MUL400 RTS
2684  CE6E               ;
2685  CE6E               ;
2686  CE6E               ;
2687  CE6E               ; DIVIDE
2688  CE6E               ;
2689  CE6E               ; RESULT=QUOTIENT ,ACCUM+1=REMAINDER
2690  CE6E               ;
2691  CE6E               ; DESTROYS A,X
2692  CE6E               ;
2693  CE6E  A9 FE        DIV254 LDA #254        ;DIVIDE BY 254
2694  CE70  2C                  .BYTE $2C      ;SKIP NEXT INSTRUCTION
2695  CE71  A9 78        DIV120 LDA #120        ;DIVIDE BY 120
2696  CE73  85 6F               STA TEMP       ;SAVE DIVISOR
2697  CE75  A2 03               LDX #3         ;SWAP ACCUM+1,2,3 WITH
2698  CE77  B5 8F        DIV100 LDA ACCUM,X     ; RESULT,1,2
2699  CE79  48                  PHA
2700  CE7A  B5 8A               LDA RESULT-1,X
2701  CE7C  95 8F               STA ACCUM,X
2702  CE7E  68                  PLA
2703  CE7F  95 8A               STA RESULT-1,X
2704  CE81  CA                  DEX
2705  CE82  D0 F3               BNE DIV100
2706  CE84  20 D9 CE            JSR ZERRES     ;RESULT=0
2707  CE87  A2 00        DIV150 LDX #0
2708  CE89  B5 90        DIV200 LDA ACCUM+1,X   ;DIVIDE BY 256
```

```
LINE# LOC   CODE        LINE


2709  CE8B  95 8F              STA ACCUM,X
2710  CE8D  E8                 INX
2711  CE8E  E0 04              CPX #4         ;DONE ?
2712  CE90  90 F7              BCC DIV200     ;NO
2713  CE92  A9 00              LDA #0         ;ZERO HI BYTE
2714  CE94  85 92              STA ACCUM+3
2715  CE96  24 6F              BIT TEMP       ;A DIV120 ?
2716  CE98  30 09              BMI DIV300     ;NO
2717  CE9A  06 8F              ASL ACCUM      ;ONLY DIVIDE BY 128
2718  CE9C  08                 PHP            ;SAVE CARRY
2719  CE9D  46 8F              LSR ACCUM      ;NORMALIZE
2720  CE9F  28                 PLP            ;RESTORE CARRY
2721  CEA0  20 E6 CE           JSR ACC200     ;2*(X/256)=X/128
2722  CEA3  20 ED CE    DIV300 JSR ADDRES      ;TOTAL A QUOTIENT
2723  CEA6  20 E5 CE           JSR ACCX2      ;A=2*A
2724  CEA9  24 6F              BIT TEMP       ;A DIV120 ?
2725  CEAB  30 03              BMI DIV400     ;NO
2726  CEAD  20 E2 CE           JSR ACCX4      ;A=4*(2*A)=8*A
2727  CEB0  A5 8F       DIV400 LDA ACCUM       ;ADD IN REMAINDER
2728  CEB2  18                 CLC
2729  CEB3  65 90              ADC ACCUM+1
2730  CEB5  85 90              STA ACCUM+1
2731  CEB7  90 06              BCC DIV500
2732  CEB9  E6 91              INC ACCUM+2
2733  CEBB  D0 02              BNE DIV500
2734  CEBD  E6 92              INC ACCUM+3
2735  CEBF  A5 92       DIV500 LDA ACCUM+3     ;TEST < 256
2736  CEC1  05 91              ORA ACCUM+2
2737  CEC3  D0 C2              BNE DIV150     ;CRUNCH SOME MORE
2738  CEC5  A5 90              LDA ACCUM+1    ;IS REMAINDER < DIVISOR
2739  CEC7  38                 SEC
2740  CEC8  E5 6F              SBC TEMP
2741  CECA  90 0C              BCC DIV700     ;YES
2742  CECC  E6 8B              INC RESULT     ;NO - FIX RESULT
2743  CECE  D0 06              BNE DIV600
2744  CED0  E6 8C              INC RESULT+1
2745  CED2  D0 02              BNE DIV600
2746  CED4  E6 8D              INC RESULT+2
2747  CED6  85 90       DIV600 STA ACCUM+1     ;NEW REMAINDER
2748  CED8  60          DIV700 RTS
2749  CED9              ;
2750  CED9              ;
2751  CED9              ;
2752  CED9              ; ZERO RESULT
2753  CED9              ;
2754  CED9  A9 00       ZERRES LDA #0
2755  CEDB  85 8B              STA RESULT
2756  CEDD  85 8C              STA RESULT+1
2757  CEDF  85 8D              STA RESULT+2
2758  CEE1  60                 RTS
2759  CEE2              ;
2760  CEE2              ;
2761  CEE2              ;
2762  CEE2              ; MULTIPLY ACCUM BY 4
2763  CEE2              ;
```

```
LINE# LOC   CODE         LINE


2764  CEE2  20 E5 CE     ACCX4  JSR ACCX2
2765  CEE5               ;
2766  CEE5               ; MULTIPLY ACCUM BY 2
2767  CEE5               ;
2768  CEE5  18           ACCX2  CLC
2769  CEE6  26 90        ACC200 ROL ACCUM+1
2770  CEE8  26 91               ROL ACCUM+2
2771  CEEA  26 92               ROL ACCUM+3
2772  CEEC  60                  RTS
2773  CEED               ;
2774  CEED               ;
2775  CEED               ;
2776  CEED               ; ADD ACCUM TO RESULT
2777  CEED               ;
2778  CEED               ; RESULT=RESULT+ACCUM+1,2,3
2779  CEED               ;
2780  CEED  18           ADDRES CLC
2781  CEEE  A2 FD               LDX #$FD
2782  CEF0  B5 8E        ADD100 LDA RESULT+3,X
2783  CEF2  75 93               ADC ACCUM+4,X
2784  CEF4  95 8E               STA RESULT+3,X
2785  CEF6  E8                  INX
2786  CEF7  D0 F7               BNE ADD100
2787  CEF9  60                  RTS
2788  CEFA               ;
2789  CEFA               ;
2790  CEFA               ;
2791  CEFA               ;
2792  CEFA               ; .END
2792  CEFA               ;
2793  CEFA                      .LIB TST2
```

```
LINE# LOC   CODE         LINE


2795  CEFA               ;
2796  CEFA               ;*******************************
2797  CEFA               ;************ L R U I N T ****
2798  CEFA               ;*******************************
2799  CEFA               ;
2800  CEFA               ; IINITIALIZE THE LRU TABLE
2801  CEFA               ;
2802  CEFA               ;*******************************
2803  CEFA               ;
2804  CEFA               LRUINT
2805  CEFA  A2 00               LDX #0
2806  CEFC               LRUILP
2807  CEFC  8A                  TXA
2808  CEFD  95 FA               STA LRUTBL,X
2809  CEFF  E8                  INX
2810  CF00  E0 04               CPX #CMDCHN
2811  CF02  D0 F8               BNE LRUILP
2812  CF04               ;
2813  CF04  A9 06               LDA #BLINDX
2814  CF06  95 FA               STA LRUTBL,X
2815  CF08  60                  RTS
2816  CF09               ;
2817  CF09               ;*******************************
2818  CF09               ;************* L R U U P D ***
2819  CF09               ;*******************************
2820  CF09               ;
2821  CF09               ; LEAST RECENTLY USED TABLE UPDATE
2822  CF09               ;
2823  CF09               ; INPUT PARAMETERS:
2824  CF09               ; LINDX - CURRENT CHANNEL
2825  CF09               ;
2826  CF09               ; OUTPUT PARAMETERS:
2827  CF09               ; LRUTBL - UPDATED
2828  CF09               ;
2829  CF09               ;*******************************
2830  CF09               ;
2831  CF09  A0 04        LRUUPD LDY #CMDCHN
2832  CF0B  A6 82               LDX LINDX
2833  CF0D  B9 FA 00     LRULP1 LDA LRUTBL,Y
2834  CF10  96 FA               STX LRUTBL,Y
2835  CF12  C5 82               CMP LINDX
2836  CF14  F0 07               BEQ LRUEXT
2837  CF16  88                  DEY
2838  CF17  30 E1               BMI LRUINT
2839  CF19  AA                  TAX
2840  CF1A  4C 0D CF            JMP LRULP1
2841  CF1D  60           LRUEXT RTS
2842  CF1E               ;
2843  CF1E               ;*******************************
2844  CF1E               ;************* D B L B U F ***
2845  CF1E               ;*******************************
2846  CF1E               ;
2847  CF1E               ; DOUBLE BUFFER
2848  CF1E               ; RTN TO SWITCH THE ACTIVE AND
2849  CF1E               ; INACTIVE BUFFERS
```

```
LINE# LOC   CODE        LINE


2850  CF1E              ;
2851  CF1E              ;*******************************
2852  CF1E              ;
2853  CF1E              DBLBUF
2854  CF1E  20 09 CF           JSR LRUUPD
2855  CF21  20 B7 DF           JSR GETINA
2856  CF24  D0 46              BNE DBL15
2857  CF26  20 D3 D1           JSR SETDRN
2858  CF29  20 8E D2           JSR GETBUF
2859  CF2C  30 48              BMI DBL30       ;NO BUFFERS
2860  CF2E  20 C2 DF           JSR PUTINA      ;STORE INACTIVE BUFF #
2861  CF31  A5 80              LDA TRACK
2862  CF33  48                 PHA
2863  CF34  A5 81              LDA SECTOR
2864  CF36  48                 PHA
2865  CF37  A9 01              LDA #1
2866  CF39  20 F6 D4           JSR DRDBYT
2867  CF3C  85 81              STA SECTOR
2868  CF3E  A9 00              LDA #0
2869  CF40  20 F6 D4           JSR DRDBYT
2870  CF43  85 80              STA TRACK
2871  CF45              ;
2872  CF45  F0 1F              BEQ DBL10
2873  CF47              ;
2874  CF47  20 25 D1           JSR TYPFIL
2875  CF4A  F0 0B              BEQ DBL05       ;IT'S REL
2876  CF4C              ;
2877  CF4C  20 AB DD           JSR TSTWRT
2878  CF4F  D0 06              BNE DBL05       ;READ AHEAD
2879  CF51              ;
2880  CF51  20 8C CF           JSR TGLBUF      ;JUST SWITCH ON WRITE
2881  CF54  4C 5D CF           JMP DBL08
2882  CF57              ;
2883  CF57              DBL05
2884  CF57  20 8C CF           JSR TGLBUF
2885  CF5A  20 57 DE           JSR RDAB
2886  CF5D              DBL08
2887  CF5D  68                 PLA
2888  CF5E  85 81              STA SECTOR
2889  CF60  68                 PLA
2890  CF61  85 80              STA TRACK
2891  CF63  4C 6F CF           JMP DBL20
2892  CF66              DBL10
2893  CF66  68                 PLA
2894  CF67  85 81              STA SECTOR
2895  CF69  68                 PLA
2896  CF6A  85 80              STA TRACK
2897  CF6C  20 8C CF    DBL15  JSR TGLBUF
2898  CF6F  20 93 DF    DBL20  JSR GETACT
2899  CF72  AA                 TAX
2900  CF73  4C 99 D5           JMP WATJOB
2901  CF76              ;
2902  CF76              ; THERE ARE NO BUFFERS TO STEAL
2903  CF76              ;
2904  CF76              DBL30
```

```
LINE# LOC   CODE        LINE


2905  CF76  A9 70              LDA #NOCHNL
2906  CF78  4C C8 C1           JMP CMDERR
2907  CF7B              ;
2908  CF7B              ;********************************
2909  CF7B              ;
2910  CF7B              DBSET
2911  CF7B  20 09 CF           JSR LRUUPD
2912  CF7E  20 B7 DF           JSR GETINA
2913  CF81  D0 08              BNE DBS10
2914  CF83  20 8E D2           JSR GETBUF
2915  CF86  30 EE              BMI DBL30      ;NO BUFFERS
2916  CF88  20 C2 DF           JSR PUTINA     ;STORE INACTIVE BUFF #
2917  CF8B              DBS10
2918  CF8B  60                 RTS
2919  CF8C              ;********************************
2920  CF8C              ;************ T G L B U F *****
2921  CF8C              ;********************************
2922  CF8C              ;
2923  CF8C              ; TOGGLE THE INACTIVE AND ACTIVE
2924  CF8C              ; BUFFERS.
2925  CF8C              ;
2926  CF8C              ; INPUT PARAMETERS:
2927  CF8C              ; LINDX - CHANNEL #
2928  CF8C              ;
2929  CF8C              ;********************************
2930  CF8C              ;
2931  CF8C  A6 82       TGLBUF LDX LINDX
2932  CF8E  B5 A7              LDA BUF0,X
2933  CF90  49 80              EOR #$80
2934  CF92  95 A7              STA BUF0,X
2935  CF94  B5 AE              LDA BUF1,X
2936  CF96  49 80              EOR #$80
2937  CF98  95 AE              STA BUF1,X
2938  CF9A  60                 RTS
2939  CF9B              ;
2940  CF9B              ;
2941  CF9B              PIBYTE
2942  CF9B  A2 12              LDX #IWSA
2943  CF9D  86 83              STX SA
2944  CF9F  20 07 D1           JSR FNDWCH
2945  CFA2  20 00 C1           JSR SETLDS
2946  CFA5  20 25 D1           JSR TYPFIL
2947  CFA8  90 05              BCC PBYTE
2948  CFAA  A9 20              LDA #OVRFLO
2949  CFAC  20 9D DD           JSR CLRFLG
2950  CFAF              PBYTE
2951  CFAF  A5 83              LDA SA
2952  CFB1  C9 0F              CMP #15
2953  CFB3  F0 23              BEQ L42
2954  CFB5  D0 08              BNE L40
2955  CFB7              ;
2956  CFB7              ; MAIN ROUTINE TO WRITE TO CHANL
2957  CFB7              ;
2958  CFB7  A5 84       PUT    LDA ORGSA       ;IS CHANL CMD OR DATA
2959  CFB9  29 8F              AND #$8F
```

```
LINE# LOC   CODE        LINE


2960  CFBB  C9 0F              CMP #15        ; <15
2961  CFBD  B0 19              BCS L42
2962  CFBF  20 25 D1    L40    JSR TYPFIL      ;DATA BYTE TO STORE
2963  CFC2  B0 05              BCS L41        ;BRANCH IF RND
2964  CFC4  A5 85              LDA DATA       ;SEQ FILE
2965  CFC6  4C 9D D1           JMP WRTBYT     ;WRITE BYTE TO CHANL
2966  CFC9              L41
2967  CFC9  D0 03              BNE L46
2968  CFCB  4C AB E0           JMP WRTREL
2969  CFCE  A5 85       L46    LDA DATA        ;RND FILE WRITE
2970  CFD0  20 F1 CF           JSR PUTBYT     ;WRITE TO CHANL
2971  CFD3  A4 82              LDY LINDX      ; PREPARE NXT BYTE
2972  CFD5  4C EE D3           JMP RNGET2
2973  CFD8  A9 04       L42    LDA #CMDCHN     ;WRITE TO CMD CHANL
2974  CFDA  85 82              STA LINDX
2975  CFDC  20 E8 D4           JSR GETPNT     ;TEST IF COMM AND BUFFER FULL
2976  CFDF  C9 2A              CMP #<CMDBUF+CMDLEN+1
2977  CFE1  F0 05              BEQ L50        ;IT IS FULL (>CMDLEN)
2978  CFE3  A5 85              LDA DATA       ;NOT FULL YET
2979  CFE5  20 F1 CF           JSR PUTBYT     ;STORE THE BYTE
2980  CFE8  A5 F8       L50    LDA EOIFLG      ;TST IF LST BYTE OF MSG
2981  CFEA  F0 01              BEQ L45        ;IT IS
2982  CFEC  60                 RTS            ;NOT YET, RETURN
2983  CFED              L45
2984  CFED  EE 55 02           INC CMDWAT     ;SET CMD WAITING FLAG
2985  CFF0  60                 RTS
2986  CFF1              ;
2987  CFF1              ; PUT .A INTO ACTIVE BUFFER OF LINDX
2988  CFF1              ;
2989  CFF1  48          PUTBYT PHA             ;SAVE .A
2990  CFF2  20 93 DF           JSR GETACT     ;GET ACTIVE BUF#
2991  CFF5  10 06              BPL PUTB1      ;BRANCH IF THERE IS ONE
2992  CFF7  68                 PLA            ;NO BUFFER ERROR
2993  CFF8  A9 61              LDA #FILNOP
2994  CFFA  4C C8 C1           JMP CMDERR     ; JMP TO ERROR ROUTINE
2995  CFFD  0A          PUTB1  ASL A           ;SAVE THE BYTE IN BUFFER
2996  CFFE  AA                 TAX
2997  CFFF  68                 PLA
2998  D000  81 99              STA (BUFTAB,X)
2999  D002  F6 99              INC BUFTAB,X   ; INC THE BUFFER POINTER
3000  D004  60                 RTS            ;Z=1 IF LAST CHAR SLOT IN BUFFER
3001  D005              ;
3002  D005              ; FIND THE ACTIVE BUFFER # (LINDX)
3003  D005              ;
3004  D005              ;
3005  D005              ;INITIALIZE DRIVES (COMMAND)
3006  D005  20 D1 C1    INTDRV JSR SIMPRS
3007  D008  20 42 D0           JSR INITDR
3008  D00B  4C 94 C1    ID20   JMP ENDCMD

3010  D00E              ; INITIALIZE DRIVE (DRVNUM)
3011  D00E              ;
3012  D00E              ITRIAL
3013  D00E  20 5D F1           JSR BAM2A
3014  D011  A8                 TAY            ;BAM LINDX FROM BUF0
```

```
LINE# LOC   CODE        LINE


3015  D012  B6 A7              LDX BUF0,Y
3016  D014  E0 FF              CPX #$FF
3017  D016  D0 14              BNE IT30       ;VALID BUFFER #
3018  D018              ;
3019  D018  48                 PHA            ;SAVE BAM-LINDX
3020  D019  20 8E D2           JSR GETBUF     ;GET A BUFFER
3021  D01C  AA                 TAX            ;TEST IT
3022  D01D  10 05              BPL IT20       ;NO ERROR
3023  D01F              ;
3024  D01F  A9 70              LDA #NOCHNL
3025  D021  20 48 E6           JSR CMDER3
3026  D024              IT20
3027  D024  68                 PLA
3028  D025  A8                 TAY            ;RESTORE BAM-LINDX
3029  D026  8A                 TXA
3030  D027  09 80              ORA #$80       ;SET INACTIVE
3031  D029  99 A7 00           STA BUF0,Y
3032  D02C              IT30
3033  D02C  8A                 TXA
3034  D02D  29 0F              AND #$F        ;STRIP TO BUF #
3035  D02F  85 F9              STA JOBNUM
3036  D031  A2 00              LDX #0
3037  D033  86 81              STX SECTOR
3038  D035  AE D4 FE           LDX DIRTRK
3039  D038  86 80              STX TRACK
3040  D03A  20 D3 D6           JSR SETH       ;SET THE BAM HEADER
3041  D03D  A9 B0              LDA #SEEK
3042  D03F  4C 8C D5           JMP DOJOB      ;DO A SEEK
3043  D042              ;
3044  D042              ; READ AND UPDATE BAM
3045  D042              INITDR
3046  D042  20 1F F1           JSR CLNBAM
3047  D045  20 13 D3           JSR CLDCHN
3048  D048  20 0E D0           JSR ITRIAL
3049  D04B  A6 7F              LDX DRVNUM
3050  D04D  A9 00              LDA #0
3051  D04F  9D 51 02           STA MDIRTY,X
3052  D052  8A                 TXA
3053  D053  0A                 ASL A
3054  D054  AA                 TAX
3055  D055  A5 16              LDA HEADER
3056  D057  95 12              STA DSKID,X
3057  D059  A5 17              LDA HEADER+1
3058  D05B  95 13              STA DSKID+1,X
3059  D05D  20 86 D5           JSR DOREAD
3060  D060  A5 F9              LDA JOBNUM
3061  D062  0A                 ASL A
3062  D063  AA                 TAX
3063  D064  A9 02              LDA #2
3064  D066  95 99              STA BUFTAB,X
3065  D068  A1 99              LDA (BUFTAB,X)
3066  D06A  A6 7F              LDX DRVNUM
3067  D06C  9D 01 01           STA DSKVER,X   ;SET UP DISK VERSION #
3068  D06F              ;
3069  D06F  A9 00              LDA #0
```

```
LINE# LOC   CODE        LINE


3070  D071  95 1C              STA WPSW,X     ;CLEAR WP SWITCH
3071  D073  95 FF              STA NODRV,X    ;CLEAR NOT ACTIVE FLAG
3072  D075              ;
3073  D075              ; COUNT THE NUMBER OF FREE BLOCKS HERE
3074  D075              ;
3075  D075              NFCALC
3076  D075  20 88 EF           JSR SETBPT
3077  D078  A0 04              LDY #4
3078  D07A  A9 00              LDA #0
3079  D07C  AA                 TAX            ;0 HI BYTE
3080  D07D              NUMF1
3081  D07D  18                 CLC
3082  D07E  71 6D              ADC (BMPNT),Y
3083  D080  90 01              BCC NUMF2
3084  D082  E8                 INX
3085  D083              NUMF2
3086  D083  C8                 INY
3087  D084  C8                 INY
3088  D085  C8                 INY
3089  D086  C8                 INY
3090  D087  C0 48              CPY #$48       ;DO NOT COUNT THE DIR
3091  D089  F0 F8              BEQ NUMF2
3092  D08B  C0 90              CPY #$90
3093  D08D  D0 EE              BNE NUMF1
3094  D08F  48                 PHA
3095  D090  8A                 TXA
3096  D091  A6 7F              LDX DRVNUM
3097  D093  9D FC 02           STA NDBH,X
3098  D096  68                 PLA
3099  D097  9D FA 02           STA NDBL,X
3100  D09A              ;
3101  D09A  60                 RTS
3102  D09B              ;
3103  D09B              ;
3104  D09B              ; START DOUBLE BUFFERING
3105  D09B              ; USE TRACK,SECTOR AS STARTING BLOACK
3106  D09B              ;
3107  D09B              STRRD
3108  D09B  20 D0 D6           JSR SETHDR
3109  D09E  20 C3 D0           JSR RDBUF
3110  D0A1  20 99 D5           JSR WATJOB
3111  D0A4  20 37 D1           JSR GETBYT
3112  D0A7  85 80              STA TRACK
3113  D0A9  20 37 D1           JSR GETBYT
3114  D0AC  85 81              STA SECTOR
3115  D0AE  60                 RTS
3116  D0AF              ;
3117  D0AF              STRDBL
3118  D0AF  20 9B D0           JSR STRRD
3119  D0B2  A5 80              LDA TRACK
3120  D0B4  D0 01              BNE STR1
3121  D0B6  60                 RTS
3122  D0B7              STR1
3123  D0B7  20 1E CF           JSR DBLBUF
3124  D0BA  20 D0 D6           JSR SETHDR
```

```
LINE# LOC   CODE        LINE


3125  D0BD  20 C3 D0           JSR RDBUF
3126  D0C0  4C 1E CF           JMP DBLBUF
3127  D0C3              ;
3128  D0C3              ; START A READ JOB ON TRACK,SECTOR
3129  D0C3              ;
3130  D0C3  A9 80       RDBUF  LDA #READ
3131  D0C5  D0 02              BNE STRTIT
3132  D0C7              ;
3133  D0C7              ; START A WRITE JOB ON TRACK,SECTOR
3134  D0C7              ;
3135  D0C7  A9 90       WRTBUF LDA #WRITE
3136  D0C9  8D 4D 02    STRTIT STA CMD
3137  D0CC  20 93 DF           JSR GETACT
3138  D0CF  AA                 TAX
3139  D0D0  20 06 D5           JSR SETLJB
3140  D0D3  8A                 TXA
3141  D0D4  48                 PHA
3142  D0D5  0A                 ASL A
3143  D0D6  AA                 TAX
3144  D0D7  A9 00              LDA #0
3145  D0D9  95 99              STA BUFTAB,X
3146  D0DB  20 25 D1           JSR TYPFIL
3147  D0DE  C9 04              CMP #4
3148  D0E0  B0 06              BCS WRTC1       ;NOT SEQUENTIAL TYPE
3149  D0E2              ;
3150  D0E2  F6 B5              INC NBKL,X
3151  D0E4  D0 02              BNE WRTC1
3152  D0E6  F6 BB              INC NBKH,X
3153  D0E8  68          WRTC1  PLA
3154  D0E9  AA                 TAX
3155  D0EA  60                 RTS
3156  D0EB              ;
3157  D0EB              ;
3158  D0EB              ;****************************
3159  D0EB              ;*
3160  D0EB              ;* FNDRCH
3161  D0EB              ;******************************
3162  D0EB              ;*
3163  D0EB              ;*
3164  D0EB  A5 83       FNDRCH LDA SA
3165  D0ED  C9 13              CMP #MAXSA+1
3166  D0EF  90 02              BCC FNDC20
3167  D0F1  29 0F              AND #$F
3168  D0F3              FNDC20
3169  D0F3  C9 0F              CMP #CMDSA
3170  D0F5  D0 02              BNE FNDC25
3171  D0F7  A9 10              LDA #$10
3172  D0F9              FNDC25
3173  D0F9  AA                 TAX
3174  D0FA  38                 SEC
3175  D0FB  BD 2B 02           LDA LINTAB,X
3176  D0FE  30 06              BMI FNDC30
3177  D100  29 0F              AND #$F
3178  D102  85 82              STA LINDX
3179  D104  AA                 TAX
```

```
LINE# LOC   CODE        LINE


3180  D105  18                 CLC
3181  D106  60          FNDC30 RTS
3182  D107              ;
3183  D107              ;
3184  D107              ;*
3185  D107              ;*
3186  D107              ;*****************************
3187  D107              ;*
3188  D107              ;* FNDWCH
3189  D107              ;*
3190  D107              ;*****************************
3191  D107              ;*
3192  D107              ;*
3193  D107  A5 83       FNDWCH LDA SA           ;CHECK IF CURRENT SECONDARY
                                                 ADDRESS
3194  D109  C9 13              CMP #MAXSA+1   ;IS VALID
3195  D10B  90 02              BCC FNDW13
3196  D10D  29 0F              AND #$0F       ;GET RELEVANT PART
3197  D10F  AA          FNDW13 TAX             ;IN X
3198  D110  BD 2B 02           LDA LINTAB,X   ;GET BUFFER #
3199  D113  A8                 TAY            ;IN Y
3200  D114  0A                 ASL A          ;MULTIPLY BY TWO
3201  D115  90 0A              BCC FNDW15     ;IF BUFFER IN USE
3202  D117  30 0A              BMI FNDW20     ;EXIT WITH ERROR
3203  D119  98          FNDW10 TYA             ;ELSE GET BUFFER NUMBER
3204  D11A  29 0F              AND #$0F       ;STRIP GARBAGE
3205  D11C  85 82              STA LINDX      ;SET IT
3206  D11E  AA                 TAX            ;GET BUFFER# IN X
3207  D11F  18                 CLC            ;FLAG NO ERROR
3208  D120  60                 RTS            ;AND EXIT
3209  D121  30 F6       FNDW15 BMI FNDW10
3210  D123  38          FNDW20 SEC             ;ELSE FLAG ERROR
3211  D124  60                 RTS            ;AND EXIT
3212  D125              ;
3213  D125              TYPFIL                  ; GET FILE TYPE
3214  D125  A6 82              LDX LINDX
3215  D127  B5 EC              LDA FILTYP,X
3216  D129  4A                 LSR A
3217  D12A  29 07              AND #7
3218  D12C  C9 04              CMP #RELTYP
3219  D12E  60                 RTS
3220  D12F              ;
3221  D12F  20 93 DF    GETPRE JSR GETACT
3222  D132  0A                 ASL A
3223  D133  AA                 TAX
3224  D134  A4 82              LDY LINDX
3225  D136  60                 RTS
3226  D137              ;
3227  D137              ; AND SET FLAG IF LAST DATA BYTE
3228  D137              ; IF LAST THEN Z=1 ELSE Z=0 ;
3229  D137  20 2F D1    GETBYT JSR GETPRE
3230  D13A  B9 44 02           LDA LSTCHR,Y
3231  D13D  F0 12              BEQ GETB1
3232  D13F  A1 99              LDA (BUFTAB,X)
3233  D141  48                 PHA
3234  D142  B5 99              LDA BUFTAB,X
```

```
LINE# LOC   CODE        LINE


3235  D144  D9 44 02           CMP LSTCHR,Y
3236  D147  D0 04              BNE GETB2
3237  D149  A9 FF              LDA #$FF
3238  D14B  95 99              STA BUFTAB,X
3239  D14D  68          GETB2  PLA
3240  D14E  F6 99              INC BUFTAB,X
3241  D150  60                 RTS
3242  D151  A1 99       GETB1  LDA (BUFTAB,X)
3243  D153  F6 99              INC BUFTAB,X
3244  D155  60                 RTS
3245  D156              ;
3246  D156              ;
3247  D156              ; READ BYTE FROM ACTIVE BUFFER
3248  D156              ; READ A CHAR FROM FILE AND READ NEXT
3249  D156              ; BLOCK OF FILE IF NEEDED.
3250  D156              ; SET CHNRDY=EOI IF END OF FILE
3251  D156              ;
3252  D156  20 37 D1    RDBYT  JSR GETBYT
3253  D159  D0 36              BNE RD3
3254  D15B  85 85              STA DATA

3256  D15D  B9 44 02    RD0    LDA LSTCHR,Y
3257  D160  F0 08              BEQ RD1
3258  D162  A9 80              LDA #EOIOUT
3259  D164  99 F2 00    RD01   STA CHNRDY,Y
3260  D167  A5 85              LDA DATA
3261  D169  60                 RTS
3262  D16A  20 1E CF    RD1    JSR DBLBUF
3263  D16D  A9 00              LDA #0
3264  D16F  20 C8 D4           JSR SETPNT
3265  D172  20 37 D1           JSR GETBYT
3266  D175  C9 00              CMP #0
3267  D177  F0 19              BEQ RD4
3268  D179  85 80              STA TRACK
3269  D17B  20 37 D1           JSR GETBYT
3270  D17E  85 81              STA SECTOR
3271  D180  20 1E CF           JSR DBLBUF
3272  D183  20 D3 D1           JSR SETDRN
3273  D186  20 D0 D6           JSR SETHDR
3274  D189  20 C3 D0           JSR RDBUF
3275  D18C  20 1E CF           JSR DBLBUF
3276  D18F  A5 85              LDA DATA
3277  D191  60          RD3    RTS
3278  D192  20 37 D1    RD4    JSR GETBYT
3279  D195  A4 82              LDY LINDX
3280  D197  99 44 02           STA LSTCHR,Y
3281  D19A  A5 85              LDA DATA
3282  D19C  60                 RTS


3285  D19D              ; WRITE A CHAR TO CHANL AND WRITE
3286  D19D              ; BUFFER OUT TO DISK IF ITS FULL
3287  D19D              ;
3288  D19D  20 F1 CF    WRTBYT JSR PUTBYT
3289  D1A0  F0 01              BEQ WRT0
```

```
LINE# LOC   CODE        LINE


3290  D1A2  60                 RTS

3292  D1A3  20 D3 D1    WRT0   JSR SETDRN
3293  D1A6  20 6C F1           JSR NXTTS
3294  D1A9  A9 00              LDA #0
3295  D1AB  20 C8 D4           JSR SETPNT
3296  D1AE  A5 80              LDA TRACK
3297  D1B0  20 F1 CF           JSR PUTBYT
3298  D1B3  A5 81              LDA SECTOR
3299  D1B5  20 F1 CF           JSR PUTBYT
3300  D1B8  20 C7 D0           JSR WRTBUF
3301  D1BB  20 1E CF           JSR DBLBUF
3302  D1BE  20 D0 D6           JSR SETHDR
3303  D1C1  A9 02              LDA #2
3304  D1C3  4C C8 D4           JMP SETPNT
3305  D1C6              ;
3306  D1C6              ; INC POINTER OF ACTIVE BUFFER
3307  D1C6              ; BY .A
3308  D1C6              ;
3309  D1C6              ;
3310  D1C6              INCPTR                ; SCOTT PATCH
3311  D1C6  85 6F       INCPNT STA TEMP
3312  D1C8  20 E8 D4           JSR GETPNT
3313  D1CB  18                 CLC
3314  D1CC  65 6F              ADC TEMP
3315  D1CE  95 99              STA BUFTAB,X
3316  D1D0  85 94              STA DIRBUF
3317  D1D2  60                 RTS
3318  D1D3              ; SET DRVNUM TO DRIVE INDICATED BY
3319  D1D3              ; LSTJOB OF ACTIVE BUFFER
3320  D1D3              ;
3321  D1D3  20 93 DF    SETDRN JSR GETACT
3322  D1D6  AA                 TAX
3323  D1D7  BD 5B 02           LDA LSTJOB,X
3324  D1DA  29 01              AND #1
3325  D1DC  85 7F              STA DRVNUM
3326  D1DE  60                 RTS
3327  D1DF              ;.END
3327  D1DF              ;
3328  D1DF                     .LIB TST3
```

```
LINE# LOC   CODE        LINE


3330  D1DF              ;
3331  D1DF              ;
3332  D1DF              ; .A=#BUFFERS NEEDED
3333  D1DF              ; SETS UP BUFFER # AND ALLOCATES LINDX
3334  D1DF              GETWCH
3335  D1DF  38                 SEC            ;SET .C=1 INDICATE WRITE
3336  D1E0  B0 01              BCS GETR2
3337  D1E2              GETRCH
3338  D1E2  18                 CLC            ;SET .C=0 INDICATE READ
3339  D1E3              GETR2
3340  D1E3  08                 PHP            ;SAVE R/W FLAG (.C)
3341  D1E4  85 6F              STA TEMP       ;SAVE #BUFS NEEDED
3342  D1E6  20 27 D2           JSR FRECHN     ;FREE ANY CHANNELS
3343  D1E9  20 7F D3           JSR FNDLNX     ;GET NEXT LINDX OPEN
3344  D1EC  85 82              STA LINDX
3345  D1EE  A6 83              LDX SA
3346  D1F0  28                 PLP
3347  D1F1  90 02              BCC GETR55
3348  D1F3  09 80       GETR52 ORA #$80
3349  D1F5  9D 2B 02    GETR55 STA LINTAB,X    ;SAVE LINDX IN LINTAB
3350  D1F8              GETBF
3351  D1F8  29 3F              AND #$3F
3352  D1FA  A8                 TAY            ;NOW GET THE BUFFERS
3353  D1FB  A9 FF              LDA #$FF
3354  D1FD  99 A7 00           STA BUF0,Y
3355  D200  99 AE 00           STA BUF1,Y
3356  D203  99 CD 00           STA SS,Y
3357  D206  C6 6F       GETR3  DEC TEMP
3358  D208  30 1C              BMI GETR4
3359  D20A  20 8E D2           JSR GETBUF
3360  D20D  10 08              BPL GETR5
3361  D20F  20 5A D2    GBERR  JSR RELBUF      ;ERROR ,REL BUFS
3362  D212  A9 70              LDA #NOCHNL
3363  D214  4C C8 C1           JMP CMDERR
3364  D217  99 A7 00    GETR5  STA BUF0,Y
3365  D21A  C6 6F              DEC TEMP
3366  D21C  30 08              BMI GETR4
3367  D21E  20 8E D2           JSR GETBUF
3368  D221  30 EC              BMI GBERR
3369  D223  99 AE 00           STA BUF1,Y
3370  D226  60          GETR4  RTS
3371  D227              ;
3372  D227              ; FREE CHANL ASSOCIATED WITH SA
3373  D227              ; FREE READ AND WRITE CHANLS
3374  D227              ; DONT FREE CHANL 15
3375  D227  A5 83       FRECHN LDA SA
3376  D229  C9 0F              CMP #$F
3377  D22B  D0 01              BNE FRECO
3378  D22D  60                 RTS

3380  D22E              FRECO
3381  D22E              FRERD
3382  D22E  A6 83       FREWRT LDX SA
3383  D230  BD 2B 02           LDA LINTAB,X
3384  D233  C9 FF              CMP #$FF
```

```
LINE# LOC   CODE        LINE


3385  D235  F0 22              BEQ FRE25
3386  D237  29 3F              AND #$3F
3387  D239  85 82              STA LINDX
3388  D23B  A9 FF              LDA #$FF
3389  D23D  9D 2B 02           STA LINTAB,X
3390  D240  A6 82              LDX LINDX
3391  D242  A9 00              LDA #0
3392  D244  95 F2              STA CHNRDY,X
3393  D246  20 5A D2           JSR RELBUF
3394  D249              ;
3395  D249              ; RELEASE THE LINDX
3396  D249              ;
3397  D249  A6 82       RELINX LDX LINDX
3398  D24B  A9 01              LDA #1
3399  D24D  CA          REL15  DEX
3400  D24E  30 03              BMI REL10
3401  D250  0A                 ASL A
3402  D251  D0 FA              BNE REL15
3403  D253  0D 56 02    REL10  ORA LINUSE      ;1=FREE 0=USED
3404  D256  8D 56 02           STA LINUSE
3405  D259  60          FRE25  RTS
3406  D25A              ; GIVEN SA , FREE ITS READ CHANL
3407  D25A              ;
3408  D25A              ;
3409  D25A              ; RELEASE BUFFERS (LINDX)
3410  D25A              ;
3411  D25A  A6 82       RELBUF LDX LINDX
3412  D25C  B5 A7              LDA BUF0,X
3413  D25E  C9 FF              CMP #$FF
3414  D260  F0 09              BEQ REL1
3415  D262  48                 PHA
3416  D263  A9 FF              LDA #$FF
3417  D265  95 A7              STA BUF0,X
3418  D267  68                 PLA
3419  D268  20 F3 D2           JSR FREBUF
3420  D26B  A6 82       REL1   LDX LINDX
3421  D26D  B5 AE              LDA BUF1,X
3422  D26F  C9 FF              CMP #$FF
3423  D271  F0 09              BEQ REL2
3424  D273  48                 PHA
3425  D274  A9 FF              LDA #$FF
3426  D276  95 AE              STA BUF1,X
3427  D278  68                 PLA
3428  D279  20 F3 D2           JSR FREBUF
3429  D27C  A6 82       REL2   LDX LINDX
3430  D27E  B5 CD              LDA SS,X
3431  D280  C9 FF              CMP #$FF
3432  D282  F0 09              BEQ REL3
3433  D284  48                 PHA
3434  D285  A9 FF              LDA #$FF
3435  D287  95 CD              STA SS,X
3436  D289  68                 PLA
3437  D28A  20 F3 D2           JSR FREBUF
3438  D28D  60          REL3   RTS
3439  D28E              ;
```

```
LINE# LOC   CODE        LINE


3440  D28E              ; GET A FREE BUFFER #
3441  D28E              ;
3442  D28E              ; REGS DESTROYED: .A .X
3443  D28E              ; OUT: .A,.X= BUF # OR $FF IF FAILED
3444  D28E              ; .N= 1 IF FAILED
3445  D28E              ;
3446  D28E              ; IF SUCCESSFUL INIT JOBS & LSTJOB
3447  D28E              ;
3448  D28E              GETBUF
3449  D28E  98                 TYA             ;SAVE .Y
3450  D28F  48                 PHA
3451  D290  A0 01              LDY #1
3452  D292  20 BA D2           JSR FNDBUF
3453  D295  10 0C              BPL GBF1        ;FOUND ONE
3454  D297              ;
3455  D297  88                 DEY
3456  D298  20 BA D2           JSR FNDBUF
3457  D29B  10 06              BPL GBF1        ;FOUND ONE
3458  D29D              ;
3459  D29D  20 39 D3           JSR STLBUF      ;STEAL ONE
3460  D2A0  AA                 TAX             ;TEST IT
3461  D2A1  30 13              BMI GBF2        ;DIDN'T FIND ONE
3462  D2A3              GBF1
3463  D2A3  B5 00              LDA JOBS,X
3464  D2A5  30 FC              BMI GBF1        ;WAIT FOR JOB FREE
3465  D2A7  A5 7F              LDA DRVNUM
3466  D2A9  95 00              STA JOBS,X      ;CLEAR JOB QUEUE
3467  D2AB  9D 5B 02           STA LSTJOB,X
3468  D2AE  8A                 TXA
3469  D2AF  0A                 ASL A
3470  D2B0  A8                 TAY
3471  D2B1  A9 02              LDA #2
3472  D2B3  99 99 00           STA BUFTAB,Y
3473  D2B6              GBF2
3474  D2B6  68                 PLA
3475  D2B7  A8                 TAY             ;RESTORE .Y
3476  D2B8  8A                 TXA             ;EXIT WITH BUF # IN .A & CC SET
3477  D2B9  60                 RTS
3478  D2BA              ;
3479  D2BA              ;
3480  D2BA              ; FIND A FREE BUF # & SET BIT IN BUFUSE
3481  D2BA              ; ALL REGS USED
3482  D2BA              ; IN: .Y= INDEX INTO BUFUSE
3483  D2BA              ; OUT: .X= BUF # OR $FF IF FAILED
3484  D2BA              ; .N= 1 IF FAILED
3485  D2BA              ;
3486  D2BA              FNDBUF
3487  D2BA  A2 07              LDX #7
3488  D2BC              FB1
3489  D2BC  B9 4F 02           LDA BUFUSE,Y    ;SEARCH BUFUSE
3490  D2BF  3D 37 F0           AND BMASK,X
3491  D2C2  F0 04              BEQ FB2         ;FOUND A FREE ONE
3492  D2C4              ;
3493  D2C4  CA                 DEX
3494  D2C5  10 F5              BPL FB1         ;UNTIL ALL BITS ARE TESTED
```

```
LINE# LOC   CODE        LINE


3495  D2C7              ;
3496  D2C7  60                 RTS
3497  D2C8              FB2                     ;SET IT USED
3498  D2C8  B9 4F 02           LDA BUFUSE,Y
3499  D2CB  5D 37 F0           EOR BMASK,X     ;SET BIT
3500  D2CE  99 4F 02           STA BUFUSE,Y
3501  D2D1  8A                 TXA
3502  D2D2  88                 DEY
3503  D2D3  30 03              BMI FB3         ;IF .Y THEN...
3504  D2D5  18                 CLC             ; ...ADD 8 MORE
3505  D2D6  69 08              ADC #8
3506  D2D8              FB3
3507  D2D8  AA                 TAX             ;LEAVE IN .X
3508  D2D9              FRI20
3509  D2D9  60                 RTS
3510  D2DA              ;
3511  D2DA              FREIAC                  ;FREE INACTIVE BUFFER
3512  D2DA  A6 82              LDX LINDX
3513  D2DC  B5 A7              LDA BUF0,X
3514  D2DE  30 09              BMI FRI10
3515  D2E0              ;
3516  D2E0  8A                 TXA
3517  D2E1  18                 CLC
3518  D2E2  69 07              ADC #MXCHNS+1
3519  D2E4  AA                 TAX
3520  D2E5  B5 A7              LDA BUF0,X
3521  D2E7  10 F0              BPL FRI20
3522  D2E9              FRI10
3523  D2E9  C9 FF              CMP #$FF
3524  D2EB  F0 EC              BEQ FRI20
3525  D2ED              ;
3526  D2ED  48                 PHA
3527  D2EE  A9 FF              LDA #$FF
3528  D2F0  95 A7              STA BUF0,X
3529  D2F2  68                 PLA
3530  D2F3              ; JMP FREBUF
3531  D2F3              ;
3532  D2F3              ;
3533  D2F3              ;
3534  D2F3  29 0F       FREBUF AND #$F
3535  D2F5  A8                 TAY
3536  D2F6  C8                 INY
3537  D2F7  A2 10              LDX #16
3538  D2F9  6E 50 02    FREB1  ROR BUFUSE+1
3539  D2FC  6E 4F 02           ROR BUFUSE
3540  D2FF  88                 DEY
3541  D300  D0 01              BNE FREB2
3542  D302  18                 CLC
3543  D303  CA          FREB2  DEX
3544  D304  10 F3              BPL FREB1
3545  D306  60                 RTS
3546  D307              ;
3547  D307              ;
3548  D307  A9 0E       CLRCHN LDA #14
3549  D309  85 83              STA SA
```

```
LINE# LOC   CODE        LINE


3550  D30B  20 27 D2    CLRC1  JSR FRECHN
3551  D30E  C6 83              DEC SA
3552  D310  D0 F9              BNE CLRC1
3553  D312  60                 RTS
3554  D313              ;
3555  D313  A9 0E       CLDCHN LDA #14
3556  D315  85 83              STA SA
3557  D317  A6 83       CLSD   LDX SA
3558  D319  BD 2B 02           LDA LINTAB,X
3559  D31C  C9 FF              CMP #$FF
3560  D31E  F0 14              BEQ CLD2
3561  D320  29 3F              AND #$3F
3562  D322  85 82              STA LINDX
3563  D324  20 93 DF           JSR GETACT
3564  D327  AA                 TAX
3565  D328  BD 5B 02           LDA LSTJOB,X
3566  D32B  29 01              AND #1
3567  D32D  C5 7F              CMP DRVNUM
3568  D32F  D0 03              BNE CLD2
3569  D331  20 27 D2           JSR FRECHN
3570  D334  C6 83       CLD2   DEC SA
3571  D336  10 DF              BPL CLSD
3572  D338  60                 RTS
3573  D339              ;
3574  D339              ;********************************
3575  D339              ;*********** S T L B U F ******
3576  D339              ;********************************
3577  D339              ;
3578  D339              ; STEAL A BUFFER
3579  D339              ;
3580  D339              ; THIS RTN SEARCHES THE CHANNELS
3581  D339              ; IN ORDER OF LEAST RECENTLY USED
3582  D339              ; AND STEALS THE FIRST INACTIVE
3583  D339              ; BUFFER IT FINDS.
3584  D339              ;
3585  D339              ; INPUT PARAMTERS:
3586  D339              ; LRUTBL - LEAST RECENTLY
3587  D339              ; USED TABLE
3588  D339              ;
3589  D339              ; OUTPUT PARAMTERS:
3590  D339              ; A <== BUFFER #
3591  D339              ;
3592  D339              ; REGISTER USAGE:
3593  D339              ; X - CHANNEL INDEX
3594  D339              ; Y - LRUTBL INDEX
3595  D339              ;
3596  D339              ;*******************************
3597  D339              ;
3598  D339              STLBUF
3599  D339  A5 6F              LDA T0
3600  D33B  48                 PHA             ;SAVE T0
3601  D33C  A0 00              LDY #0
3602  D33E              STL05
3603  D33E  B6 FA              LDX LRUTBL,Y
3604  D340  B5 A7              LDA BUF0,X
```

```
LINE# LOC   CODE         LINE


3605  D342  10 04               BPL STL10       ;ACTIVE
3606  D344               ;
3607  D344  C9 FF               CMP #$FF
3608  D346  D0 16               BNE STL30       ;IT'S INACTIVE
3609  D348               STL10
3610  D348  8A                  TXA
3611  D349  18                  CLC
3612  D34A  69 07               ADC #MXCHNS+1
3613  D34C  AA                  TAX
3614  D34D  B5 A7               LDA BUF0,X
3615  D34F  10 04               BPL STL20
3616  D351  C9 FF               CMP #$FF
3617  D353  D0 09               BNE STL30
3618  D355               STL20
3619  D355  C8                  INY
3620  D356  C0 05               CPY #MXCHNS-1
3621  D358  90 E4               BCC STL05
3622  D35A  A2 FF               LDX #$FF        ;SET FAILURE
3623  D35C  D0 1C               BNE STL60       ;BRA
3624  D35E               STL30                   ;STEAL THE BUFFER IF NO ERROR
3625  D35E  86 6F               STX T0
3626  D360  29 3F               AND #$3F
3627  D362  AA                  TAX
3628  D363               STL40
3629  D363  B5 00               LDA JOBS,X
3630  D365  30 FC               BMI STL40       ;WAIT TILL DONE
3631  D367  C9 02               CMP #2          ;ERRORS?
3632  D369  90 08               BCC STL50       ;OK
3633  D36B  A6 6F               LDX T0
3634  D36D  E0 07               CPX #MXCHNS+1
3635  D36F  90 D7               BCC STL10       ;CHECK OPPOSITE SLOT
3636  D371  B0 E2               BCS STL20       ;CHECK ANOTHER CHANNEL
3637  D373               ;
3638  D373               ; FOUND ONE, SO LETS STEAL IT
3639  D373               ;
3640  D373               STL50
3641  D373  A4 6F               LDY T0
3642  D375  A9 FF               LDA #$FF
3643  D377  99 A7 00            STA BUF0,Y      ;CLEAR SLOT
3644  D37A               STL60
3645  D37A  68                  PLA
3646  D37B  85 6F               STA T0
3647  D37D  8A                  TXA             ;BUF # IN .A & SET CC'S
3648  D37E  60                  RTS
3649  D37F               ;
3650  D37F               ;
3651  D37F               ;
3652  D37F               ; FIND A FREE LINDX TO USE
3653  D37F               ; MARK AS USED IN LINUSE
3654  D37F               ;
3655  D37F  A0 00        FNDLNX LDY #0
3656  D381  A9 01               LDA #1
3657  D383  2C 56 02     FND10  BIT LINUSE       ;1=FREE 0=USED
3658  D386  D0 09               BNE FND30
3659  D388  C8                  INY
```

```
LINE# LOC   CODE        LINE


3660  D389  0A                 ASL A
3661  D38A  D0 F7              BNE FND10
3662  D38C  A9 70              LDA #NOCHNL    ;NO FREE LINDX AVAILABLE
3663  D38E  4C C8 C1           JMP CMDERR
3664  D391  49 FF       FND30  EOR #$FF        ;TOGGLE BIT MASK
3665  D393  2D 56 02           AND LINUSE     ;MARK BIT USED
3666  D396  8D 56 02           STA LINUSE
3667  D399  98                 TYA            ;RETURN LINDX IN .A
3668  D39A  60                 RTS
3669  D39B              ;.END
3669  D39B              ;
3670  D39B                     .LIB TST4
```

```
LINE# LOC   CODE        LINE


3672  D39B              ;
3673  D39B              ; GET NEXT CHAR FROM A CHANL
3674  D39B              ;
3675  D39B              GBYTE                  ;GET THROUGH INTERNAL CHANNEL
3676  D39B  20 EB D0           JSR FNDRCH
3677  D39E  20 00 C1           JSR SETLDS
3678  D3A1  20 AA D3           JSR GET
3679  D3A4  A6 82              LDX LINDX
3680  D3A6  BD 3E 02           LDA CHNDAT,X
3681  D3A9  60                 RTS
3682  D3AA              ;
3683  D3AA              ;
3684  D3AA  A6 82       GET    LDX LINDX
3685  D3AC  20 25 D1           JSR TYPFIL     ;GET FILE TYPE
3686  D3AF  D0 03              BNE GET00
3687  D3B1              ;
3688  D3B1  4C 20 E1           JMP RDREL
3689  D3B4              GET00
3690  D3B4  A5 83              LDA SA
3691  D3B6  C9 0F              CMP #$F
3692  D3B8  F0 5A              BEQ GETERC
3693  D3BA              ;
3694  D3BA  B5 F2              LDA CHNRDY,X   ;WAS LAST CHAR JUST SENT
3695  D3BC  29 08              AND #EOISND    ;JUST SENT EOI
3696  D3BE  D0 13              BNE GET1       ;NOPE NOT THIS TIME
3697  D3C0              ;
3698  D3C0  20 25 D1           JSR TYPFIL
3699  D3C3  C9 07              CMP #DIRTYP
3700  D3C5  D0 07              BNE GET0       ;NOT DIRECT TYPE
3701  D3C7              ;
3702  D3C7  A9 89              LDA #RNDRDY    ;DIRECT FILE REMAINS ACTIVE
3703  D3C9  95 F2              STA CHNRDY,X   ;TALKER LISTENER NOEOI
3704  D3CB  4C DE D3           JMP RNDGET     ;PREPARE THE NEXT CHAR
3705  D3CE              ;
3706  D3CE  A9 00       GET0   LDA #NOTRDY     ;LST CHAR SENT,NOT READY
3707  D3D0  95 F2              STA CHNRDY,X
3708  D3D2  60                 RTS
3709  D3D3              ;
3710  D3D3              ;
3711  D3D3  A5 83       GET1   LDA SA          ;TEST IF A LOAD
3712  D3D5  F0 32              BEQ GET6       ;ITS A LOAD SA OF 0
3713  D3D7              ;
3714  D3D7  20 25 D1    GET2   JSR TYPFIL      ;TST FOR RND FILE
3715  D3DA  C9 04              CMP #4
3716  D3DC  90 22              BCC SEQGET     ;SEQ FILE CHAR GET
3717  D3DE              ;
3718  D3DE  20 2F D1    RNDGET JSR GETPRE      ;DIRECT FILE GET
3719  D3E1  B5 99              LDA BUFTAB,X
3720  D3E3  D9 44 02           CMP LSTCHR,Y   ;UP TO LST CHAR YET
3721  D3E6  D0 04              BNE RNGET1     ;NOT YET
3722  D3E8              ;
3723  D3E8  A9 00              LDA #0         ;READ THE WHOLE THING
3724  D3EA  95 99              STA BUFTAB,X   ;WRAP PNTR TO 0
3725  D3EC              RNGET1
3726  D3EC  F6 99              INC BUFTAB,X   ;GET THE NEXT CHAR
```

```
LINE# LOC   CODE         LINE


3727  D3EE               RNGET2
3728  D3EE  A1 99               LDA (BUFTAB,X)
3729  D3F0               RNGET4
3730  D3F0  99 3E 02            STA CHNDAT,Y   ;SAVE CHAR IN
3731  D3F3  B5 99               LDA BUFTAB,X
3732  D3F5  D9 44 02            CMP LSTCHR,Y
3733  D3F8  D0 05               BNE RNGET3
3734  D3FA               ;
3735  D3FA  A9 81               LDA #RNDEOI
3736  D3FC  99 F2 00            STA CHNRDY,Y   ;THIS IS LAST CHAR
3737  D3FF  60           RNGET3 RTS             ;SEND EOI WITH IT
3738  D400               ;
3739  D400               ;
3740  D400  20 56 D1     SEQGET JSR RDBYT        ;READ THE NEXT BYTE
3741  D403  A6 82        GET3   LDX LINDX
3742  D405  9D 3E 02            STA CHNDAT,X   ;STORE IN CHNDAT
3743  D408  60                  RTS
3744  D409               ;
3745  D409               ;
3746  D409  AD 54 02     GET6   LDA DIRLST      ;TEST IF DIR LSTING
3747  D40C  F0 F2               BEQ SEQGET     ;NO, ITS NOT
3748  D40E  20 B5 ED            JSR GETDIR     ;YES IT IS,GET DIR CHAR
3749  D411  4C 03 D4            JMP GET3
3750  D414               ;
3751  D414               GETERC
3752  D414  20 E8 D4            JSR GETPNT
3753  D417  C9 D4               CMP #<ERRBUF-1
3754  D419  D0 18               BNE GE10
3755  D41B  A5 95               LDA DIRBUF+1
3756  D41D  C9 02               CMP #>ERRBUF
3757  D41F  D0 12               BNE GE10
3758  D421               ;
3759  D421  A9 0D               LDA #CR
3760  D423  85 85               STA DATA
3761  D425  20 23 C1            JSR ERROFF
3762  D428  A9 00               LDA #0
3763  D42A  20 C1 E6            JSR ERRTS0
3764  D42D  C6 A5               DEC CB+2
3765  D42F  A9 80               LDA #EOIOUT
3766  D431  D0 12               BNE GE30       ;(JUMP)
3767  D433               ;
3768  D433               GE10
3769  D433  20 37 D1            JSR GETBYT
3770  D436  85 85               STA DATA
3771  D438  D0 09               BNE GE20
3772  D43A               GE15
3773  D43A  A9 D4               LDA #<ERRBUF-1
3774  D43C  20 C8 D4            JSR SETPNT
3775  D43F  A9 02               LDA #>ERRBUF
3776  D441  95 9A               STA BUFTAB+1,X
3777  D443               GE20
3778  D443  A9 88               LDA #RDYTLK
3779  D445               GE30
3780  D445  85 F7               STA CHNRDY+ERRCHN
3781  D447  A5 85               LDA DATA
```

```
LINE# LOC   CODE         LINE


3782  D449  8D 43 02            STA CHNDAT+ERRCHN
3783  D44C  60                  RTS
3784  D44D                 ;
3785  D44D                 ;
3786  D44D                 ; READ NEXT BUFFER OF A FILE
3787  D44D                 ; FOLLOW LINKS IN FIRST TWO BYTES
3788  D44D                 ; END OF FILE IF 1ST BYTE=0
3789  D44D                 ; 2ND CHAR LENGTH
3790  D44D                 ;
3791  D44D  20 93 DF     NXTBUF JSR GETACT
3792  D450  0A                  ASL A
3793  D451  AA                  TAX
3794  D452  A9 00               LDA #0
3795  D454  95 99               STA BUFTAB,X
3796  D456  A1 99               LDA (BUFTAB,X)
3797  D458  F0 05               BEQ NXTB1
3798  D45A  D6 99               DEC BUFTAB,X
3799  D45C  4C 56 D1            JMP RDBYT
3800  D45F  60           NXTB1  RTS
3801  D460                 ;
3802  D460                 ; DIRECT BLOCK READ
3803  D460                 ;
3804  D460                 ;
3805  D460  A9 80        DRTRD  LDA #READ
3806  D462  D0 02               BNE DRT
3807  D464                 ; DIRECT BLOCK WRITE
3808  D464                 ;
3809  D464                 ; DIRECT BLOCK WRITE
3810  D464  A9 90        DRTWRT LDA #WRITE
3811  D466  05 7F        DRT    ORA DRVNUM
3812  D468  8D 4D 02            STA CMD
3813  D46B  A5 F9               LDA JOBNUM
3814  D46D  20 D3 D6            JSR SETH
3815  D470  A6 F9               LDX JOBNUM
3816  D472  4C 93 D5            JMP DOIT2
3817  D475                 ; OPEN INTERNAL READ CHANL (SA=16)
3818  D475                 ;
3819  D475               OPNIRD
3820  D475  A9 01               LDA #1
3821  D477               OPNTYP
3822  D477  8D 4A 02            STA TYPE
3823  D47A  A9 11               LDA #IRSA
3824  D47C  85 83               STA SA
3825  D47E  20 46 DC            JSR OPNRCH
3826  D481  A9 02               LDA #2
3827  D483  4C C8 D4            JMP SETPNT
3828  D486                 ; OPEN INTERNAL WRITE CHANL (SA=16)
3829  D486                 ;
3830  D486                 ;
3831  D486  A9 12        OPNIWR LDA #IWSA
3832  D488  85 83               STA SA
3833  D48A  4C DA DC            JMP OPNWCH
3834  D48D                 ;
3835  D48D                 ; ALLOCATE NEXT DIRECTORY BLOCK ON 18
3836  D48D                 ; AND MARK AS USED IN BAM
```

```
LINE# LOC   CODE         LINE


3837  D48D               ;
3838  D48D  20 3B DE     NXDRBK JSR CURBLK
3839  D490  A9 01               LDA #1
3840  D492  85 6F               STA TEMP
3841  D494  A5 69               LDA SECINC
3842  D496  48                  PHA
3843  D497  A9 03               LDA #3          ; INCR SECTOR BY 3 IN DIRECTORY
3844  D499  85 69               STA SECINC
3845  D49B  20 7B F1            JSR NXTDS
3846  D49E  68                  PLA
3847  D49F  85 69               STA SECINC
3848  D4A1  A9 00               LDA #0
3849  D4A3  20 C8 D4            JSR SETPNT
3850  D4A6  A5 80               LDA TRACK
3851  D4A8  20 F1 CF            JSR PUTBYT
3852  D4AB  A5 81               LDA SECTOR
3853  D4AD  20 F1 CF            JSR PUTBYT
3854  D4B0  20 C7 D0            JSR WRTBUF
3855  D4B3  20 99 D5            JSR WATJOB
3856  D4B6  A9 00               LDA #0
3857  D4B8  20 C8 D4            JSR SETPNT
3858  D4BB  20 F1 CF     NXDB1  JSR PUTBYT
3859  D4BE  D0 FB               BNE NXDB1
3860  D4C0  20 F1 CF            JSR PUTBYT
3861  D4C3  A9 FF               LDA #$FF
3862  D4C5  4C F1 CF            JMP PUTBYT
3863  D4C8               ;
3864  D4C8               ;
3865  D4C8               ; .A=NEW PNTR VALUE
3866  D4C8               ;
3867  D4C8  85 6F        SETPNT STA TEMP
3868  D4CA  20 93 DF            JSR GETACT
3869  D4CD  0A                  ASL A
3870  D4CE  AA                  TAX
3871  D4CF  B5 9A               LDA BUFTAB+1,X
3872  D4D1  85 95               STA DIRBUF+1
3873  D4D3  A5 6F               LDA TEMP
3874  D4D5  95 99               STA BUFTAB,X
3875  D4D7  85 94               STA DIRBUF
3876  D4D9  60                  RTS
3877  D4DA               ;
3878  D4DA               ;
3879  D4DA               ;
3880  D4DA               ; FREE THE INTERNAL CHANL (SA=16)
3881  D4DA               ;
3882  D4DA  A9 11        FREICH LDA #IRSA
3883  D4DC  85 83               STA SA
3884  D4DE  20 27 D2            JSR FRECHN
3885  D4E1  A9 12               LDA #IWSA
3886  D4E3  85 83               STA SA
3887  D4E5  4C 27 D2            JMP FRECHN
3888  D4E8               ;
3889  D4E8               ; READ THE ACTIVE BUFFER POINTER
3890  D4E8               ;
3891  D4E8  20 93 DF     GETPNT JSR GETACT
```

```
LINE# LOC   CODE        LINE


3892  D4EB              SETDIR
3893  D4EB  0A          GP1    ASL A
3894  D4EC  AA                 TAX
3895  D4ED  B5 9A              LDA BUFTAB+1,X
3896  D4EF  85 95              STA DIRBUF+1
3897  D4F1  B5 99              LDA BUFTAB,X
3898  D4F3  85 94              STA DIRBUF
3899  D4F5  60                 RTS
3900  D4F6              ;
3901  D4F6              ; DIRECT READ BYTE, .A=BYTE# TO READ
3902  D4F6              ;
3903  D4F6  85 71       DRDBYT STA TEMP+2
3904  D4F8  20 93 DF           JSR GETACT
3905  D4FB  AA                 TAX
3906  D4FC  BD 2F FF           LDA BUFIND,X
3907  D4FF  85 72              STA TEMP+3
3908  D501  A0 00              LDY #0
3909  D503  B1 71              LDA (TEMP+2),Y
3910  D505  60                 RTS
3911  D506              ; INDEX TABLE CONTAINING HIGH
3912  D506              ; BYTE ADDRESS OF BUFFERS
3913  D506              ;
3914  D506              ;.END
3914  D506              ;
3915  D506                     .LIB JOBSSF
```

```
LINE# LOC   CODE        LINE


3917  D506              ;
3918  D506              ; USE LASTJOB FOR DRIVE #
3919  D506              ; CMD IS USED FOR JOB COMMAND
3920  D506              ;
3921  D506              SETLJB
3922  D506  BD 5B 02           LDA LSTJOB,X
3923  D509  29 01              AND #1
3924  D50B  0D 4D 02           ORA CMD
3925  D50E              ;
3926  D50E              ; SET JOB UP AND CHECK T&S
3927  D50E              ; .A=COMMAND FOR JOBS
3928  D50E              ; .X=JOB NUMBER
3929  D50E              ;
3930  D50E              SETJOB
3931  D50E  48                 PHA
3932  D50F  86 F9              STX JOBNUM
3933  D511  8A                 TXA
3934  D512              ;ASL A ;4/12*****************
3935  D512              ;ASL A
3936  D512  0A                 ASL A
3937  D513  AA                 TAX
3938  D514  B5 07              LDA HDRS+1,X   ;4/12***********
3939  D516  8D 4D 02           STA CMD        ;SAVE SECTOR
3940  D519  B5 06              LDA HDRS,X     ;4/12***********
3941  D51B  F0 2D              BEQ TSERR
3942  D51D              ;
3943  D51D  CD 26 FF           CMP MAXTRK
3944  D520  B0 28              BCS TSERR      ;TRACK TOO LARGE
3945  D522              ;
3946  D522  AA                 TAX
3947  D523  68                 PLA            ;CHECK FOR WRITE
3948  D524  48                 PHA
3949  D525  29 F0              AND #$F0
3950  D527  C9 90              CMP #WRITE
3951  D529  D0 4F              BNE SJB1       ;NOT WRITE,SKIP CHECK
3952  D52B  68                 PLA
3953  D52C  48                 PHA
3954  D52D  4A                 LSR A
3955  D52E  B0 05              BCS SJB2       ;DRIVE 1
3956  D530              ;
3957  D530  AD 01 01           LDA DSKVER     ;GET VERSION #
3958  D533  90 03              BCC SJB3
3959  D535              SJB2
3960  D535  AD 02 01           LDA DSKVER+1   ;GET DRIVE 1 VER#
3961  D538              SJB3
3962  D538  F0 05              BEQ SJB4       ;NO # IS OK, TOO
3963  D53A  CD 24 FF           CMP VERNUM
3964  D53D  D0 33              BNE VNERR      ;NOT SAME VERNUM #
3965  D53F              ;
3966  D53F              SJB4
3967  D53F  8A                 TXA            ;RESTORE TRACK #
3968  D540  20 99 F2           JSR MAXSEC
3969  D543  CD 4D 02           CMP CMD
3970  D546  F0 02              BEQ TSERR
3971  D548  B0 30              BCS SJB1       ;SECTOR IS OK!
```

```
LINE# LOC   CODE        LINE


3972  D54A              ;
3973  D54A              ;
3974  D54A              ;ILLEGAL TRACK AND SECTOR
3975  D54A              ;
3976  D54A              TSERR
3977  D54A  20 52 D5           JSR HED2TS
3978  D54D              TSER1
3979  D54D  A9 66              LDA #BADTS
3980  D54F  4C 45 E6           JMP CMDER2
3981  D552              ;
3982  D552              ;
3983  D552              HED2TS
3984  D552  A5 F9              LDA JOBNUM
3985  D554              ;ASL A ;4/12*************
3986  D554              ;ASL A
3987  D554  0A                 ASL A
3988  D555  AA                 TAX
3989  D556  B5 06              LDA HDRS,X     ;4/12***********
3990  D558  85 80              STA TRACK
3991  D55A  B5 07              LDA HDRS+1,X   ;4/12***********
3992  D55C  85 81              STA SECTOR
3993  D55E  60                 RTS
3994  D55F              ;
3995  D55F              ;
3996  D55F              TSCHK
3997  D55F  A5 80              LDA TRACK
3998  D561  F0 EA              BEQ TSER1
3999  D563  CD 26 FF           CMP MAXTRK
4000  D566  B0 E5              BCS TSER1
4001  D568              ;
4002  D568  20 99 F2           JSR MAXSEC
4003  D56B  C5 81              CMP SECTOR
4004  D56D  F0 DE              BEQ TSER1
4005  D56F  90 DC              BCC TSER1
4006  D571  60                 RTS
4007  D572              ;
4008  D572              VNERR
4009  D572  20 52 D5           JSR HED2TS
4010  D575  A9 73              LDA #CBMV2     ;WRITE TO WRONG VERSION
4011  D577  4C 45 E6           JMP CMDER2
4012  D57A              ;
4013  D57A              SJB1
4014  D57A  A6 F9              LDX JOBNUM
4015  D57C  68                 PLA
4016  D57D  8D 4D 02           STA CMD
4017  D580  95 00              STA JOBS,X
4018  D582  9D 5B 02           STA LSTJOB,X
4019  D585  60                 RTS
4020  D586              ;
4021  D586              ;
4022  D586              ; DO JOB IN .A, SET UP ERROR COUNT
4023  D586              ; AND LSTJOB. RETURN WHEN JOB DONE OK
4024  D586              ; JMP TO ERROR IF ERROR RETURNS
4025  D586              ;
4026  D586              DOREAD
```

```
LINE# LOC   CODE        LINE


4027  D586  A9 80              LDA #READ
4028  D588  D0 02              BNE DOJOB      ;BRA
4029  D58A              DOWRIT
4030  D58A  A9 90              LDA #WRITE
4031  D58C              DOJOB
4032  D58C  05 7F              ORA DRVNUM
4033  D58E  A6 F9              LDX JOBNUM
4034  D590              ;
4035  D590  8D 4D 02    DOIT   STA CMD
4036  D593  AD 4D 02    DOIT2  LDA CMD
4037  D596  20 0E D5           JSR SETJOB
4038  D599              ; JMP WATJOB
4039  D599              ;
4040  D599              ; WAIT UNTIL JOB(.X) IS DONE
4041  D599              ; RETURN WHEN DONE
4042  D599              ;
4043  D599  20 A6 D5    WATJOB JSR TSTJOB
4044  D59C  B0 FB              BCS WATJOB
4045  D59E  48                 PHA            ;CLR JOBRTN FLAG
4046  D59F  A9 00              LDA #0
4047  D5A1  8D 98 02           STA JOBRTN
4048  D5A4  68                 PLA
4049  D5A5  60                 RTS
4050  D5A6              ;
4051  D5A6              ;
4052  D5A6              ; TEST IF JOB(.X) IS DONE YET
4053  D5A6              ; IF NOT DONE RETURN
4054  D5A6              ; IF OK THEN RETURN ELSE REDO IT
4055  D5A6              ;
4056  D5A6  B5 00       TSTJOB LDA JOBS,X
4057  D5A8  30 1A              BMI NOTYET
4058  D5AA  C9 02              CMP #2
4059  D5AC  90 14              BCC OK
4060  D5AE              ;
4061  D5AE  C9 08              CMP #8         ;CHECK FOR WP SWITCH ON
4062  D5B0  F0 08              BEQ TJ10
4063  D5B2              ;
4064  D5B2  C9 0B              CMP #11        ;CHECK FOR ID MISMATCH
4065  D5B4  F0 04              BEQ TJ10
4066  D5B6              ;
4067  D5B6  C9 0F              CMP #$F        ;CHECK FOR NODRIVE
4068  D5B8  D0 0C              BNE RECOV
4069  D5BA  2C 98 02    TJ10   BIT JOBRTN
4070  D5BD  30 03              BMI OK
4071  D5BF  4C 3F D6           JMP QUIT2
4072  D5C2              ;
4073  D5C2  18          OK     CLC             ;C=0 FINISHED OK OR QUIT
4074  D5C3  60                 RTS
4075  D5C4              ;
4076  D5C4  38          NOTYET SEC             ;C=1 NOT YET
4077  D5C5  60                 RTS
4078  D5C6              ;
4079  D5C6              ;
4080  D5C6              RECOV
4081  D5C6  98                 TYA            ;SAVE .Y
```

```
LINE# LOC   CODE        LINE


4082  D5C7  48                 PHA
4083  D5C8  A5 7F              LDA DRVNUM      ;SAVE DRIVE #
4084  D5CA  48                 PHA
4085  D5CB  BD 5B 02           LDA LSTJOB,X
4086  D5CE  29 01              AND #1
4087  D5D0  85 7F              STA DRVNUM      ;SET ACTIVE DRIVE #
4088  D5D2              ;
4089  D5D2  A8                 TAY
4090  D5D3  B9 19 FF           LDA LEDMSK,Y
4091  D5D6  8D 6D 02           STA ERLED
4092  D5D9              ;
4093  D5D9  20 A6 D6           JSR DOREC
4094  D5DC  C9 02              CMP #2
4095  D5DE  B0 03              BCS REC01
4096  D5E0  4C 6D D6           JMP REC95
4097  D5E3              REC01
4098  D5E3              ;
4099  D5E3  BD 5B 02           LDA LSTJOB,X    ;ORIGINAL JOB
4100  D5E6  29 F0              AND #$F0        ;MASK JOB CODE
4101  D5E8  48                 PHA             ;SAVE IT
4102  D5E9  C9 90              CMP #WRITE
4103  D5EB  D0 07              BNE REC0        ;NOT A WRITE
4104  D5ED              ;
4105  D5ED  A5 7F              LDA DRVNUM
4106  D5EF  09 B8              ORA #SECSEK     ;REPLACE W/ SECTOR SEEK...
4107  D5F1  9D 5B 02           STA LSTJOB,X    ;... DURING RECOVERY
4108  D5F4              REC0
4109  D5F4  24 6A              BIT REVCNT
4110  D5F6  70 39              BVS REC5
4111  D5F8  A9 00              LDA #0
4112  D5FA  8D 99 02           STA EPTR        ;CLR OFFSET TABLE PTR
4113  D5FD  8D 9A 02           STA TOFF        ;CLR TOTAL OFFSET
4114  D600              REC1
4115  D600  AC 99 02           LDY EPTR
4116  D603  AD 9A 02           LDA TOFF
4117  D606  38                 SEC
4118  D607  F9 2A FF           SBC OFFSET,Y
4119  D60A  8D 9A 02           STA TOFF        ;KEEP TRACK OF ALL OFFSETS
4120  D60D  B9 2A FF           LDA OFFSET,Y
4121  D610  20 76 D6           JSR HEDOFF
4122  D613              ;
4123  D613  EE 99 02           INC EPTR        ;BUMP TABLE PTR
4124  D616  20 A6 D6           JSR DOREC       ;DO THE RECOVERY
4125  D619  C9 02              CMP #2          ;ERROR CODE < 2?
4126  D61B  90 08              BCC REC3        ;JOB WORKED
4127  D61D              ;
4128  D61D  AC 99 02           LDY EPTR
4129  D620  B9 2A FF           LDA OFFSET,Y
4130  D623  D0 DB              BNE REC1        ;NULL INDICATES END
4131  D625              REC3
4132  D625  AD 9A 02           LDA TOFF
4133  D628  20 76 D6           JSR HEDOFF
4134  D62B  B5 00              LDA JOBS,X
4135  D62D  C9 02              CMP #2
4136  D62F  90 2B              BCC REC9        ;NO ERROR
```

```
LINE# LOC   CODE        LINE


4137  D631              REC5
4138  D631  24 6A              BIT REVCNT      ;CHECK BUMP-ON FLAG
4139  D633  10 0F              BPL REC7        ;NO BUMP
4140  D635              ;
4141  D635              QUIT
4142  D635  68                 PLA
4143  D636  C9 90              CMP #WRITE      ;CHECK ORIGINAL JOB
4144  D638  D0 05              BNE QUIT2
4145  D63A              ;
4146  D63A  05 7F              ORA DRVNUM
4147  D63C  9D 5B 02           STA LSTJOB,X    ;MUST RESTORE ORIGINAL
4148  D63F              QUIT2
4149  D63F  B5 00              LDA JOBS,X      ;.A= ERROR #
4150  D641  20 0A E6           JSR ERROR
4151  D644              REC7
4152  D644  68                 PLA
4153  D645  2C 98 02           BIT JOBRTN
4154  D648  30 23              BMI REC95       ;RETURN JOB ERROR
4155  D64A  48                 PHA
4156  D64B              ;
4157  D64B              ; DO THE BUMP
4158  D64B  A9 C0              LDA #BUMP
4159  D64D  05 7F              ORA DRVNUM
4160  D64F  95 00              STA JOBS,X
4161  D651              REC8
4162  D651  B5 00              LDA JOBS,X
4163  D653  30 FC              BMI REC8        ;WAIT
4164  D655              ;
4165  D655  20 A6 D6           JSR DOREC       ;TRY ONE LAST SET
4166  D658  C9 02              CMP #2
4167  D65A  B0 D9              BCS QUIT        ;IT CLEARLY AIN'T GONNA WORK
4168  D65C              REC9
4169  D65C  68                 PLA             ;CHECK ORIGINAL JOB FOR WRITE
4170  D65D  C9 90              CMP #WRITE
4171  D65F  D0 0C              BNE REC95       ;ORIGINAL JOB WORKED
4172  D661              ;
4173  D661  05 7F              ORA DRVNUM
4174  D663  9D 5B 02           STA LSTJOB,X    ;SET WRITE JOB BACK
4175  D666  20 A6 D6           JSR DOREC       ;TRY LAST SET OF WRITES
4176  D669  C9 02              CMP #2          ;CHECK ERROR CODE
4177  D66B  B0 D2              BCS QUIT2       ;ERROR
4178  D66D              REC95
4179  D66D  68                 PLA
4180  D66E  85 7F              STA DRVNUM      ;RESTORE DRIVE #
4181  D670  68                 PLA
4182  D671  A8                 TAY             ;RESTORE .Y
4183  D672  B5 00              LDA JOBS,X
4184  D674  18                 CLC             ;OK!
4185  D675  60                 RTS
4186  D676              ;
4187  D676              HEDOFF                  ;.A=OFFSET
4188  D676  C9 00              CMP #0
4189  D678  F0 18              BEQ HOF3        ;NO OFFSET
4190  D67A  30 0C              BMI HOF2        ;STEPS ARE INWARD
4191  D67C              HOF1
```

```
LINE# LOC   CODE        LINE


4192  D67C  A0 01              LDY #1          ;STEP OUT 1 TRACK
4193  D67E  20 93 D6           JSR MOVHED
4194  D681  38                 SEC
4195  D682  E9 01              SBC #1
4196  D684  D0 F6              BNE HOF1        ;NOT FINISHED
4197  D686  F0 0A              BEQ HOF3
4198  D688              HOF2
4199  D688  A0 FF              LDY #$FF        ;STEP IN 1 TRACK
4200  D68A  20 93 D6           JSR MOVHED
4201  D68D  18                 CLC
4202  D68E  69 01              ADC #1
4203  D690  D0 F6              BNE HOF2        ; NOT FINISHED
4204  D692              HOF3
4205  D692  60                 RTS
4206  D693              ;
4207  D693              MOVHED
4208  D693  48                 PHA             ;SAVE .A
4209  D694  98                 TYA             ;PUT PHASE IN .A
4210  D695  A4 7F              LDY DRVNUM
4211  D697  99 FE 02           STA PHASE,Y
4212  D69A              MH10
4213  D69A  D9 FE 02           CMP PHASE,Y
4214  D69D  F0 FB              BEQ MH10        ;WAIT FOR CONTROLLER
4215  D69F              ;TO CHANGE IT
4216  D69F  A9 00              LDA #0
4217  D6A1  99 FE 02           STA PHASE,Y     ;CLEAR IT OUT
4218  D6A4  68                 PLA             ;RESTORE
4219  D6A5  60                 RTS
4220  D6A6              ;
4221  D6A6              ;
4222  D6A6              DOREC                   ;DO LAST JOB RECOVERY
4223  D6A6  A5 6A              LDA REVCNT      ;RE-TRY JOB REVCNT...
4224  D6A8  29 3F              AND #$3F        ;...# OF TIMES
4225  D6AA  A8                 TAY
4226  D6AB              DOREC1
4227  D6AB  AD 6D 02           LDA ERLED
4228  D6AE  4D 00 1C           EOR LEDPRT
4229  D6B1  8D 00 1C           STA LEDPRT
4230  D6B4  BD 5B 02           LDA LSTJOB,X    ;SET LAST JOB
4231  D6B7  95 00              STA JOBS,X
4232  D6B9              DOREC2
4233  D6B9  B5 00              LDA JOBS,X      ;WAIT
4234  D6BB  30 FC              BMI DOREC2
4235  D6BD  C9 02              CMP #2
4236  D6BF  90 03              BCC DOREC3      ;IT WORKED
4237  D6C1              ;
4238  D6C1  88                 DEY
4239  D6C2  D0 E7              BNE DOREC1      ;KEEP TRYING
4240  D6C4              DOREC3
4241  D6C4  48                 PHA
4242  D6C5  AD 6D 02           LDA ERLED       ;LEAVE DRIVE LED ON
4243  D6C8  0D 00 1C           ORA LEDPRT
4244  D6CB  8D 00 1C           STA LEDPRT
4245  D6CE  68                 PLA
4246  D6CF  60                 RTS             ;FINISHED
```

```
LINE# LOC   CODE        LINE


4247  D6D0              ;
4248  D6D0              ; SET HEADER OF ACTIVE BUFFER OF THE
4249  D6D0              ; CURRENT LINDX TO TRACK,SECTOR,ID
4250  D6D0              ;
4251  D6D0  20 93 DF    SETHDR JSR GETACT
4252  D6D3              SETH
4253  D6D3              ;ASL A ;4/12*****************
4254  D6D3              ;ASL A
4255  D6D3  0A                 ASL A
4256  D6D4  A8                 TAY
4257  D6D5  A5 80              LDA TRACK
4258  D6D7  99 06 00           STA HDRS,Y     ;4/12*********** ;SET TRACK
                                               4259  D6DA  A5 81
                                               LDA SECTOR
4260  D6DC  99 07 00           STA HDRS+1,Y   ;4/12*********** ;SET SECTOR
4261  D6DF  A5 7F              LDA DRVNUM     ;GET PROPER ID(DRVNUM)
4262  D6E1  0A                 ASL A
4263  D6E2  AA                 TAX
4264  D6E3              ;LDA DSKID,X
4265  D6E3              ;STA HDRS,Y ;4/12***********
4266  D6E3              ;LDA DSKID+1,X
4267  D6E3              ;STA HDRS+1,Y ;4/12***********
4268  D6E3  60                 RTS
4269  D6E4              ;
4270  D6E4              ; .END
4270  D6E4              ;
4271  D6E4                     .LIB ADDFIL
```

```
LINE# LOC   CODE        LINE


4273  D6E4              ; ADD FILE TO DIRECTORY
4274  D6E4              ;
4275  D6E4  A5 83       ADDFIL LDA SA           ; SAVE VARIABLES
4276  D6E6  48                 PHA
4277  D6E7  A5 82              LDA LINDX
4278  D6E9  48                 PHA
4279  D6EA  A5 81              LDA SECTOR
4280  D6EC  48                 PHA
4281  D6ED  A5 80              LDA TRACK
4282  D6EF  48                 PHA
4283  D6F0  A9 11              LDA #IRSA
4284  D6F2  85 83              STA SA
4285  D6F4  20 3B DE           JSR CURBLK      ; USE LAST ACCESSED SEARCH
4286  D6F7  AD 4A 02           LDA TYPE
4287  D6FA  48                 PHA
4288  D6FB  A5 E2              LDA FILDRV
4289  D6FD  29 01              AND #1
4290  D6FF  85 7F              STA DRVNUM
4291  D701  A6 F9              LDX JOBNUM
4292  D703  5D 5B 02           EOR LSTJOB,X
4293  D706  4A                 LSR A
4294  D707  90 0C              BCC AF08        ; SAME DRIVE AS REQUIRED
4295  D709              ;
4296  D709  A2 01              LDX #1
4297  D70B  8E 92 02           STX DELIND      ; LOOK FOR DELETED ENTRY
4298  D70E  20 AC C5           JSR SRCHST
4299  D711  F0 1D              BEQ AF15        ; ALL FULL, NEW SECTOR
4300  D713  D0 28              BNE AF20        ; FOUND ONE
4301  D715              ;
4302  D715  AD 91 02    AF08   LDA DELSEC
4303  D718  F0 0C              BEQ AF10        ; DELETED ENTRY NOT LOCATED
4304  D71A  C5 81              CMP SECTOR
4305  D71C  F0 1F              BEQ AF20        ; SECTOR IS RESIDENT
4306  D71E  85 81              STA SECTOR
4307  D720  20 60 D4           JSR DRTRD       ; READ SECTOR IN
4308  D723  4C 3D D7           JMP AF20
4309  D726              ;
4310  D726  A9 01       AF10   LDA #1           ; FIND DELETED ENTRY
4311  D728  8D 92 02           STA DELIND
4312  D72B  20 17 C6           JSR SEARCH
4313  D72E  D0 0D              BNE AF20
4314  D730  20 8D D4    AF15   JSR NXDRBK       ; ALL FULL, NEW SECTOR
4315  D733  A5 81              LDA SECTOR
4316  D735  8D 91 02           STA DELSEC
4317  D738  A9 02              LDA #2
4318  D73A  8D 92 02           STA DELIND
4319  D73D  AD 92 02    AF20   LDA DELIND
4320  D740  20 C8 D4           JSR SETPNT
4321  D743  68                 PLA
4322  D744  8D 4A 02           STA TYPE        ; SET TYPE
4323  D747  C9 04              CMP #RELTYP
4324  D749  D0 02              BNE AF25
4325  D74B  09 80              ORA #$80
4326  D74D              AF25
4327  D74D  20 F1 CF           JSR PUTBYT
```

```
LINE# LOC   CODE        LINE


4328  D750  68                 PLA
4329  D751  8D 80 02           STA FILTRK     ; ...TABLE & ENTRY
4330  D754  20 F1 CF           JSR PUTBYT
4331  D757  68                 PLA
4332  D758  8D 85 02           STA FILSEC     ; SET SECTOR LINK IN...
4333  D75B  20 F1 CF           JSR PUTBYT     ; ...TABLE & ENTRY
4334  D75E  20 93 DF           JSR GETACT
4335  D761  A8                 TAY
4336  D762  AD 7A 02           LDA FILTBL
4337  D765  AA                 TAX
4338  D766  A9 10              LDA #16
4339  D768  20 6E C6           JSR TRNAME     ; TRANSFER NAME
4340  D76B  A0 10              LDY #16
4341  D76D  A9 00              LDA #0         ; CLEAR # OF BLOCKS &...
4342  D76F  91 94       AF30   STA (DIRBUF),Y  ; ...& REPLACE LINKS
4343  D771  C8                 INY
4344  D772  C0 1B              CPY #27
4345  D774  90 F9              BCC AF30
4346  D776  AD 4A 02           LDA TYPE       ; A RELATIVE FILE ?
4347  D779  C9 04              CMP #RELTYP
4348  D77B  D0 13              BNE AF50       ; NO
4349  D77D  A0 10              LDY #16        ; YES
4350  D77F  AD 59 02           LDA TRKSS      ; GET SS TRACK
4351  D782  91 94              STA (DIRBUF),Y ; PUT IN DIRECTORY
4352  D784  C8                 INY
4353  D785  AD 5A 02           LDA SECSS      ; GET SS SECTOR
4354  D788  91 94              STA (DIRBUF),Y ; PUT IN
4355  D78A  C8                 INY
4356  D78B  AD 58 02           LDA REC        ; GET RECORD SIZE
4357  D78E  91 94              STA (DIRBUF),Y
4358  D790  20 64 D4    AF50   JSR DRTWRT      ; WRITE IT OUT
4359  D793  68                 PLA
4360  D794  85 82              STA LINDX
4361  D796  AA                 TAX
4362  D797  68                 PLA
4363  D798  85 83              STA SA
4364  D79A  AD 91 02           LDA DELSEC
4365  D79D  85 D8              STA ENTSEC
4366  D79F  9D 60 02           STA DSEC,X
4367  D7A2  AD 92 02           LDA DELIND
4368  D7A5  85 DD              STA ENTIND
4369  D7A7  9D 66 02           STA DIND,X
4370  D7AA  AD 4A 02           LDA TYPE
4371  D7AD  85 E7              STA PATTYP
4372  D7AF  A5 7F              LDA DRVNUM
4373  D7B1  85 E2              STA FILDRV
4374  D7B3  60                 RTS
4375  D7B4              ;
4376  D7B4              ; .END
4376  D7B4              ;
4377  D7B4                     .LIB OPEN
```

```
LINE# LOC   CODE        LINE


4379  D7B4              ;OPEN CHANNEL FROM IEEE
4380  D7B4              ; PARSES THE INPUT STRING THAT IS
4381  D7B4              ; SENT AS AN OPEN DATA CHANNEL,
4382  D7B4              ; LOAD, OR SAVE. CHANNELS ARE ALLOCATED
4383  D7B4              ; AND THE DIRECTORY IS SEARCHED FOR
4384  D7B4              ; THE FILENAME CONTAINED IN THE STRING.

4386  D7B4              OPEN
4387  D7B4  A5 83              LDA SA
4388  D7B6  8D 4C 02           STA TEMPSA
4389  D7B9  20 B3 C2           JSR CMDSET     ; INITIATE CMD PTRS
4390  D7BC  8E 2A 02           STX CMDNUM
4391  D7BF  AE 00 02           LDX CMDBUF
4392  D7C2  AD 4C 02           LDA TEMPSA
4393  D7C5  D0 2C              BNE OP021
4394  D7C7  E0 2A              CPX #'*        ; LOAD LAST?
4395  D7C9  D0 28              BNE OP021
4396  D7CB  A5 7E              LDA PRGTRK
4397  D7CD  F0 4D              BEQ OP0415     ; NO LAST PROG, INIT 0

4399  D7CF              OP02                   ; LOAD LAST PROGRAM
4400  D7CF  85 80              STA TRACK
4401  D7D1  AD 6E 02           LDA PRGDRV
4402  D7D4  85 7F              STA DRVNUM
4403  D7D6  85 E2              STA FILDRV
4404  D7D8  A9 02              LDA #PRGTYP
4405  D7DA  85 E7              STA PATTYP
4406  D7DC  AD 6F 02           LDA PRGSEC
4407  D7DF  85 81              STA SECTOR
4408  D7E1  20 00 C1           JSR SETLDS     ; MAKE SURE LED GETS TURNED ON!!
4409  D7E4  20 46 DC           JSR OPNRCH
4410  D7E7  A9 04              LDA #PRGTYP+PRGTYP
4411  D7E9  05 7F              ORA DRVNUM
4412  D7EB  A6 82       ENDRD  LDX LINDX
4413  D7ED  99 EC 00           STA FILTYP,Y
4414  D7F0  4C 94 C1           JMP ENDCMD

4416  D7F3  E0 24       OP021  CPX #'$'
4417  D7F5  D0 1E              BNE OP041
4418  D7F7  AD 4C 02           LDA TEMPSA     ; LOAD DIRECTORY
4419  D7FA  D0 03              BNE OP04
4420  D7FC  4C 55 DA           JMP LOADIR

4422  D7FF  20 D1 C1    OP04   JSR SIMPRS      ; OPEN DIR AS SEQ FILE
4423  D802  AD D4 FE           LDA DIRTRK
4424  D805  85 80              STA TRACK
4425  D807  A9 00              LDA #0
4426  D809  85 81              STA SECTOR
4427  D80B  20 46 DC           JSR OPNRCH
4428  D80E  A5 7F              LDA DRVNUM
4429  D810  09 02              ORA #SEQTYP+SEQTYP
4430  D812  4C EB D7           JMP ENDRD

4432  D815  E0 23       OP041  CPX #'#'        ; "#" IS DIRECT ACCESS
4433  D817  D0 12              BNE OP042
```

```
LINE# LOC   CODE         LINE


4434  D819  4C 84 CB            JMP OPNBLK

4436  D81C  A9 02        OP0415 LDA #PRGTYP      ; PROGRAM TYPE
4437  D81E  8D 96 02            STA TYPFLG
4438  D821  A9 00               LDA #0
4439  D823  85 7F               STA DRVNUM
4440  D825  8D 8E 02            STA LSTDRV
4441  D828  20 42 D0            JSR INITDR

4443  D82B               OP042
4444  D82B  20 E5 C1            JSR PRSCLN      ; LOOK FOR ":"
4445  D82E  D0 04               BNE OP049
4446  D830  A2 00               LDX #0
4447  D832  F0 0C               BEQ OP20        ; BRA
4448  D834               OP049
4449  D834  8A                  TXA
4450  D835  F0 05               BEQ OP10
4451  D837  A9 30        OP05   LDA #BADSYN      ; SOMETHING AMISS
4452  D839  4C C8 C1            JMP CMDERR
4453  D83C  88           OP10   DEY              ; BACK UP TO ":"
4454  D83D  F0 01               BEQ OP20        ; 1ST CHAR IS ":"
4455  D83F  88                  DEY
4456  D840  8C 7A 02     OP20   STY FILTBL       ; SAVE FILENAME PTR
4457  D843  A9 8D               LDA #$8D        ; LOOK FOR CR-SHIFTED
4458  D845  20 68 C2            JSR PARSE

4460  D848  E8                  INX
4461  D849  8E 78 02            STX F2CNT
4462  D84C  20 12 C3            JSR ONEDRV
4463  D84F  20 CA C3            JSR OPTSCH
4464  D852  20 9D C4            JSR FFST        ; LOOK FOR FILE ENTRY
4465  D855  A2 00               LDX #0
4466  D857  8E 58 02            STX REC
4467  D85A  8E 97 02            STX MODE        ; READ MODE
4468  D85D  8E 4A 02            STX TYPE        ; DELETED
4469  D860  E8                  INX
4470  D861  EC 77 02            CPX F1CNT
4471  D864  B0 10               BCS OP40        ; NO PARAMETERS
4472  D866               ;
4473  D866  20 09 DA            JSR CKTM        ; CHECK FOR TYPE & MODE
4474  D869  E8                  INX
4475  D86A  EC 77 02            CPX F1CNT
4476  D86D  B0 07               BCS OP40        ; ONLY ONE PARAMETER
4477  D86F               ;
4478  D86F  C0 04               CPY #RELTYP
4479  D871  F0 3E               BEQ OP60        ; SET RECORD SIZE
4480  D873               ;
4481  D873  20 09 DA            JSR CKTM        ; SET TYPE/MODE
4482  D876               OP40
4483  D876  AE 4C 02            LDX TEMPSA
4484  D879  86 83               STX SA          ; SET SA BACK
4485  D87B  E0 02               CPX #2
4486  D87D  B0 12               BCS OP45        ; NOT LOAD OR SAVE
4487  D87F               ;
4488  D87F  8E 97 02            STX MODE        ; MODE=SA
```

```
LINE# LOC   CODE        LINE


4489  D882  A9 40              LDA #$40
4490  D884  8D F9 02           STA WBAM
4491  D887  AD 4A 02           LDA TYPE
4492  D88A  D0 1B              BNE OP50       ; TYPE FROM PARM
4493  D88C              ;
4494  D88C  A9 02              LDA #PRGTYP
4495  D88E  8D 4A 02           STA TYPE       ; USE PRG
4496  D891              OP45
4497  D891  AD 4A 02           LDA TYPE
4498  D894  D0 11              BNE OP50       ; TYPE FROM PARM
4499  D896              ;
4500  D896  A5 E7              LDA PATTYP
4501  D898  29 07              AND #TYPMSK
4502  D89A  8D 4A 02           STA TYPE       ; TYPE FROM FILE
4503  D89D              ;
4504  D89D  AD 80 02           LDA FILTRK
4505  D8A0  D0 05              BNE OP50       ; YES, IT EXISTS
4506  D8A2              ;
4507  D8A2  A9 01              LDA #SEQTYP
4508  D8A4  8D 4A 02           STA TYPE       ; DEFAULT IS SEQ
4509  D8A7              OP50
4510  D8A7  AD 97 02           LDA MODE
4511  D8AA  C9 01              CMP #WTMODE
4512  D8AC  F0 18              BEQ OP75       ; GO WRITE
4513  D8AE              ;
4514  D8AE  4C 40 D9           JMP OP90
4515  D8B1              ;
4516  D8B1              OP60
4517  D8B1  BC 7A 02           LDY FILTBL,X   ; GET RECORD SIZE
4518  D8B4  B9 00 02           LDA CMDBUF,Y
4519  D8B7  8D 58 02           STA REC
4520  D8BA  AD 80 02           LDA FILTRK
4521  D8BD  D0 B7              BNE OP40       ; IT'S HERE, READ
4522  D8BF              ;
4523  D8BF  A9 01              LDA #WTMODE    ; USE WRITE TO OPEN
4524  D8C1  8D 97 02           STA MODE
4525  D8C4  D0 B0              BNE OP40       ; (BRA)
4526  D8C6              ;
4527  D8C6              OP75
4528  D8C6  A5 E7              LDA PATTYP
4529  D8C8  29 80              AND #$80
4530  D8CA  AA                 TAX
4531  D8CB  D0 14              BNE OP81
4532  D8CD  A9 20       OP77   LDA #$20        ; OPEN WRITE
4533  D8CF  24 E7              BIT PATTYP
4534  D8D1  F0 06              BEQ OP80
4535  D8D3  20 B6 C8           JSR DELDIR     ; CREATED
4536  D8D6  4C E3 D9           JMP OPWRIT

4538  D8D9  AD 80 02    OP80   LDA FILTRK
4539  D8DC  D0 03              BNE OP81
4540  D8DE  4C E3 D9           JMP OPWRIT     ; NOT FOUND, OK!
4541  D8E1  AD 00 02    OP81   LDA CMDBUF
4542  D8E4  C9 40              CMP #'@'       ; CHECK FOR REPLACE
4543  D8E6  F0 0D              BEQ OP82
```

```
LINE# LOC   CODE        LINE


4544  D8E8  8A                 TXA
4545  D8E9  D0 05              BNE OP815
4546  D8EB  A9 63              LDA #FLEXST
4547  D8ED  4C C8 C1           JMP CMDERR
4548  D8F0              OP815
4549  D8F0  A9 33              LDA #BADFN
4550  D8F2  4C C8 C1           JMP CMDERR

4552  D8F5              ;********* CHECK FOR BUG HERE******
4553  D8F5              OP82
4554  D8F5  A5 E7              LDA PATTYP     ;REPLACE
4555  D8F7  29 07              AND #$07
4556  D8F9  CD 4A 02           CMP TYPE
4557  D8FC  D0 67              BNE OP115
4558  D8FE  C9 04              CMP #RELTYP
4559  D900  F0 63              BEQ OP115
4560  D902              ;
4561  D902              ;
4562  D902  20 DA DC           JSR OPNWCH
4563  D905  A5 82              LDA LINDX
4564  D907  8D 70 02           STA WLINDX
4565  D90A  A9 11              LDA #IRSA      ; INTERNAL CHAN
4566  D90C  85 83              STA SA
4567  D90E  20 EB D0           JSR FNDRCH
4568  D911  AD 94 02           LDA INDEX
4569  D914  20 C8 D4           JSR SETPNT
4570  D917  A0 00              LDY #0
4571  D919  B1 94              LDA (DIRBUF),Y
4572  D91B  09 20              ORA #$20       ; SET REPLACE BIT
4573  D91D  91 94              STA (DIRBUF),Y
4574  D91F              ;
4575  D91F  A0 1A              LDY #26
4576  D921  A5 80              LDA TRACK
4577  D923  91 94              STA (DIRBUF),Y
4578  D925  C8                 INY
4579  D926  A5 81              LDA SECTOR
4580  D928  91 94              STA (DIRBUF),Y

4582  D92A  AE 70 02           LDX WLINDX
4583  D92D  A5 D8              LDA ENTSEC
4584  D92F  9D 60 02           STA DSEC,X
4585  D932  A5 DD              LDA ENTIND
4586  D934  9D 66 02           STA DIND,X
4587  D937  20 3B DE           JSR CURBLK
4588  D93A  20 64 D4           JSR DRTWRT
4589  D93D  4C EF D9           JMP OPFIN
4590  D940              ;*********************************
4591  D940              ;

4593  D940  AD 80 02    OP90   LDA FILTRK      ; OPEN READ (& LOAD)
4594  D943  D0 05              BNE OP100
4595  D945              OP95
4596  D945  A9 62              LDA #FLNTFD    ; TRACK NOT RECORDED
4597  D947  4C C8 C1           JMP CMDERR     ; NOT FOUND
4598  D94A              OP100
```

```
LINE# LOC   CODE        LINE


4599  D94A  AD 97 02           LDA MODE
4600  D94D  C9 03              CMP #MDMODE
4601  D94F  F0 0B              BEQ OP110
4602  D951  A9 20              LDA #$20
4603  D953  24 E7              BIT PATTYP
4604  D955  F0 05              BEQ OP110
4605  D957  A9 60              LDA #FILOPN
4606  D959  4C C8 C1           JMP CMDERR
4607  D95C  A5 E7       OP110  LDA PATTYP
4608  D95E  29 07              AND #TYPMSK    ; TYPE IS IN INDEX TABLE
4609  D960  CD 4A 02           CMP TYPE
4610  D963  F0 05              BEQ OP120
4611  D965  A9 64       OP115  LDA #MISTYP     ; TYPE MISMATCH
4612  D967  4C C8 C1           JMP CMDERR
4613  D96A              OP120                  ; EVERYTHING IS OK!
4614  D96A  A0 00              LDY #0
4615  D96C  8C 79 02           STY F2PTR
4616  D96F  AE 97 02           LDX MODE
4617  D972  E0 02              CPX #APMODE
4618  D974  D0 1A              BNE OP125
4619  D976  C9 04              CMP #RELTYP
4620  D978  F0 EB              BEQ OP115
4621  D97A              ;
4622  D97A  B1 94              LDA (DIRBUF),Y
4623  D97C  29 4F              AND #$4F
4624  D97E  91 94              STA (DIRBUF),Y
4625  D980  A5 83              LDA SA
4626  D982  48                 PHA
4627  D983  A9 11              LDA #IRSA
4628  D985  85 83              STA SA
4629  D987  20 3B DE           JSR CURBLK
4630  D98A  20 64 D4           JSR DRTWRT
4631  D98D  68                 PLA
4632  D98E  85 83              STA SA
4633  D990              OP125
4634  D990  20 A0 D9           JSR OPREAD
4635  D993  AD 97 02           LDA MODE
4636  D996  C9 02              CMP #APMODE
4637  D998  D0 55              BNE OPFIN
4638  D99A              ;
4639  D99A  20 2A DA           JSR APPEND
4640  D99D  4C 94 C1           JMP ENDCMD
4641  D9A0              ;
4642  D9A0              ;**********************
4643  D9A0              OPREAD
4644  D9A0  A0 13              LDY #19
4645  D9A2  B1 94              LDA (DIRBUF),Y
4646  D9A4  8D 59 02           STA TRKSS
4647  D9A7  C8                 INY
4648  D9A8  B1 94              LDA (DIRBUF),Y
4649  D9AA  8D 5A 02           STA SECSS
4650  D9AD  C8                 INY
4651  D9AE  B1 94              LDA (DIRBUF),Y
4652  D9B0  AE 58 02           LDX REC
4653  D9B3  8D 58 02           STA REC
```

```
LINE# LOC   CODE        LINE


4654  D9B6  8A                 TXA
4655  D9B7  F0 0A              BEQ OP130
4656  D9B9  CD 58 02           CMP REC
4657  D9BC  F0 05              BEQ OP130
4658  D9BE  A9 50              LDA #NOREC
4659  D9C0  20 C8 C1           JSR CMDERR
4660  D9C3              OP130
4661  D9C3  AE 79 02           LDX F2PTR
4662  D9C6  BD 80 02           LDA FILTRK,X
4663  D9C9  85 80              STA TRACK
4664  D9CB  BD 85 02           LDA FILSEC,X
4665  D9CE  85 81              STA SECTOR
4666  D9D0  20 46 DC           JSR OPNRCH
4667  D9D3  A4 82              LDY LINDX       ; OPEN A READ CHNL
4668  D9D5  AE 79 02           LDX F2PTR
4669  D9D8  B5 D8              LDA ENTSEC,X
4670  D9DA  99 60 02           STA DSEC,Y
4671  D9DD  B5 DD              LDA ENTIND,X
4672  D9DF  99 66 02           STA DIND,Y
4673  D9E2  60                 RTS
4674  D9E3              ;

4676  D9E3              OPWRIT
4677  D9E3  A5 E2              LDA FILDRV
4678  D9E5  29 01              AND #1
4679  D9E7  85 7F              STA DRVNUM
4680  D9E9  20 DA DC           JSR OPNWCH
4681  D9EC  20 E4 D6           JSR ADDFIL      ; ADD TO DIRECTORY
4682  D9EF              OPFIN
4683  D9EF  A5 83              LDA SA
4684  D9F1  C9 02              CMP #2
4685  D9F3  B0 11              BCS OPF1
4686  D9F5              ;
4687  D9F5  20 3E DE           JSR GETHDR
4688  D9F8  A5 80              LDA TRACK
4689  D9FA  85 7E              STA PRGTRK
4690  D9FC              ;
4691  D9FC  A5 7F              LDA DRVNUM
4692  D9FE  8D 6E 02           STA PRGDRV
4693  DA01              ;
4694  DA01  A5 81              LDA SECTOR
4695  DA03  8D 6F 02           STA PRGSEC
4696  DA06              OPF1
4697  DA06  4C 99 C1           JMP ENDSAV
4698  DA09              ;
4699  DA09              CKTM
4700  DA09  BC 7A 02           LDY FILTBL,X    ; GET PTR
4701  DA0C  B9 00 02           LDA CMDBUF,Y    ; GET CHAR
4702  DA0F  A0 04              LDY #NMODES
4703  DA11              CKM1
4704  DA11  88                 DEY
4705  DA12  30 08              BMI CKM2        ; NO VALID MODE
4706  DA14              ;
4707  DA14  D9 01 FF           CMP MODLST,Y
4708  DA17  D0 F8              BNE CKM1
```

```
LINE# LOC   CODE        LINE


4709  DA19  8C 97 02           STY MODE       ; MODE FOUND
4710  DA1C              CKM2
4711  DA1C  A0 05              LDY #NTYPES
4712  DA1E              CKT1
4713  DA1E  88                 DEY
4714  DA1F  30 08              BMI CKT2       ; NO VALID TYPE
4715  DA21              ;
4716  DA21  D9 05 FF           CMP TPLST,Y
4717  DA24  D0 F8              BNE CKT1
4718  DA26  8C 4A 02           STY TYPE       ; TYPE FOUND
4719  DA29              CKT2
4720  DA29  60                 RTS
4721  DA2A              ;
4722  DA2A              APPEND
4723  DA2A  20 39 CA           JSR GCBYTE
4724  DA2D  A9 80              LDA #LRF
4725  DA2F  20 A6 DD           JSR TSTFLG
4726  DA32  F0 F6              BEQ APPEND
4727  DA34              ;
4728  DA34  20 95 DE           JSR RDLNK
4729  DA37  A6 81              LDX SECTOR
4730  DA39  E8                 INX
4731  DA3A  8A                 TXA
4732  DA3B  D0 05              BNE AP30
4733  DA3D  20 A3 D1           JSR WRT0       ; GET ANOTHER BLOCK
4734  DA40  A9 02              LDA #2
4735  DA42              AP30
4736  DA42  20 C8 D4           JSR SETPNT
4737  DA45  A6 82              LDX LINDX
4738  DA47  A9 01              LDA #RDYLST
4739  DA49  95 F2              STA CHNRDY,X
4740  DA4B  A9 80              LDA #$80       ; CHNL BIT
4741  DA4D  05 82              ORA LINDX
4742  DA4F  A6 83              LDX SA
4743  DA51  9D 2B 02           STA LINTAB,X
4744  DA54  60                 RTS
4745  DA55              ;
4746  DA55              ;
4747  DA55              ; LOAD DIRECTORY
4748  DA55              LOADIR
4749  DA55  A9 0C              LDA #LDCMD
4750  DA57  8D 2A 02           STA CMDNUM
4751  DA5A  A9 00              LDA #0         ; LOAD ONLY DRIVE ZERO
4752  DA5C  AE 74 02           LDX CMDSIZ
4753  DA5F  CA                 DEX
4754  DA60  F0 0B              BEQ LD02

4756  DA62  CA          LD01   DEX             ; LOAD BY NAME
4757  DA63  D0 21              BNE LD03
4758  DA65  AD 01 02           LDA CMDBUF+1
4759  DA68  20 BD C3           JSR TST0V1
4760  DA6B  30 19              BMI LD03

4762  DA6D              LD02                   ; LOAD DIR WITH A STAR
4763  DA6D  85 E2              STA FILDRV
```

```
LINE# LOC   CODE        LINE


4764  DA6F  EE 77 02           INC F1CNT
4765  DA72  EE 78 02           INC F2CNT
4766  DA75  EE 7A 02           INC FILTBL
4767  DA78  A9 80              LDA #$80
4768  DA7A  85 E7              STA PATTYP
4769  DA7C  A9 2A              LDA #'*
4770  DA7E  8D 00 02           STA CMDBUF     ; COVER BOTH CASES
4771  DA81  8D 01 02           STA CMDBUF+1
4772  DA84  D0 18              BNE LD10       ; (BRANCH)

4774  DA86              LD03
4775  DA86  20 E5 C1           JSR PRSCLN
4776  DA89  D0 05              BNE LD05       ; FOUND ":"
4777  DA8B              ;SEARCH BY NAME ON BOTH DRIVES
4778  DA8B  20 DC C2           JSR CMDRST
4779  DA8E  A0 03              LDY #3
4780  DA90  88          LD05   DEY
4781  DA91  88                 DEY
4782  DA92  8C 7A 02           STY FILTBL

4784  DA95  20 00 C2           JSR TC35       ; PARSE & SET TABLES
4785  DA98  20 98 C3           JSR FS1SET
4786  DA9B  20 20 C3           JSR ALLDRS

4788  DA9E  20 CA C3    LD10   JSR OPTSCH      ; NEW DIRECTORY
4789  DAA1  20 B7 C7           JSR NEWDIR
4790  DAA4  20 9D C4           JSR FFST
4791  DAA7  20 EC EC    LD20   JSR STDIR       ; START DIRECTORY
4792  DAAA  20 37 D1           JSR GETBYT     ; SET 1ST BYTE
4793  DAAD  A6 82              LDX LINDX
4794  DAAF  9D 3E 02           STA CHNDAT,X
4795  DAB2  A5 7F              LDA DRVNUM
4796  DAB4  8D 8E 02           STA LSTDRV
4797  DAB7  09 04              ORA #4
4798  DAB9  95 EC              STA FILTYP,X
4799  DABB  A9 00              LDA #0
4800  DABD  85 A3              STA BUFTAB+CBPTR
4801  DABF  60                 RTS

4803  DAC0              ;.END
4803  DAC0              ;
4804  DAC0                     .LIB CLOSE
```

```
LINE# LOC   CODE         LINE


4806  DAC0               ;CLOSE THE FILE ASSOCIATED WITH SA
4807  DAC0               ;
4808  DAC0               CLOSE
4809  DAC0  A9 00               LDA #0
4810  DAC2  8D F9 02            STA WBAM
4811  DAC5  A5 83               LDA SA
4812  DAC7  D0 0B               BNE CLS10       ; DIRECTORY CLOSE
4813  DAC9  A9 00               LDA #0
4814  DACB  8D 54 02            STA DIRLST      ;CLEAR DIR LIST
4815  DACE  20 27 D2            JSR FRECHN
4816  DAD1               CLS05
4817  DAD1  4C DA D4            JMP FREICH
4818  DAD4               CLS10
4819  DAD4  C9 0F               CMP #$F
4820  DAD6  F0 14               BEQ CLSALL      ; CLOSE CMD CHANL
4821  DAD8  20 02 DB            JSR CLSCHN      ;CLOSE CHANNEL
4822  DADB  A5 83               LDA SA
4823  DADD  C9 02               CMP #2
4824  DADF  90 F0               BCC CLS05
4825  DAE1               ;
4826  DAE1  AD 6C 02            LDA ERWORD
4827  DAE4  D0 03               BNE CLS15       ; LAST COMMAND HAD AN ERROR

4828  DAE6  4C 94 C1            JMP ENDCMD
4829  DAE9               CLS15
4830  DAE9  4C AD C1            JMP SCREN1
4831  DAEC               ;
4832  DAEC               CLSALL
4833  DAEC  A9 0E               LDA #14
4834  DAEE  85 83               STA SA
4835  DAF0               CLS20
4836  DAF0  20 02 DB            JSR CLSCHN
4837  DAF3  C6 83               DEC SA
4838  DAF5  10 F9               BPL CLS20
4839  DAF7  AD 6C 02            LDA ERWORD
4840  DAFA  D0 03               BNE CLS25       ; LAST COMMAND HAD AN ERROR

4841  DAFC  4C 94 C1            JMP ENDCMD
4842  DAFF               CLS25
4843  DAFF  4C AD C1            JMP SCREN1
4844  DB02               ;
4845  DB02               CLSCHN
4846  DB02  A6 83               LDX SA
4847  DB04  BD 2B 02            LDA LINTAB,X
4848  DB07  C9 FF               CMP #$FF
4849  DB09  D0 01               BNE CLSC28
4850  DB0B  60                  RTS
4851  DB0C               CLSC28
4852  DB0C  29 0F               AND #$F
4853  DB0E  85 82               STA LINDX
4854  DB10               ;
4855  DB10  20 25 D1            JSR TYPFIL
4856  DB13  C9 07               CMP #DIRTYP
4857  DB15  F0 0F               BEQ CLSC30      ;DIRECT CHANNEL
4858  DB17  C9 04               CMP #RELTYP
4859  DB19  F0 11               BEQ CLSREL
4860  DB1B               ;
```

```
LINE# LOC   CODE         LINE


4861  DB1B  20 07 D1            JSR FNDWCH      ;LOOK FOR WRITE CHANNEL
4862  DB1E  B0 09               BCS CLSC31
4863  DB20               ;
4864  DB20  20 62 DB            JSR CLSWRT      ;CLOSE SEQ WRITE
4865  DB23  20 A5 DB            JSR CLSDIR      ;CLOSE DIRECTORY
4866  DB26               CLSC30
4867  DB26  20 42 EF            JSR MAPOUT      ;WRITE BAM
4868  DB29               CLSC31
4869  DB29  4C 27 D2            JMP FRECHN
4870  DB2C               ;
4871  DB2C               CLSREL
4872  DB2C  20 F1 DD            JSR SCRUB
4873  DB2F  20 1E CF            JSR DBLBUF
4874  DB32  20 CB E1            JSR SSEND
4875  DB35  A6 D5               LDX SSNUM
4876  DB37  86 73               STX T4
4877  DB39  E6 73               INC T4
4878  DB3B  A9 00               LDA #0
4879  DB3D  85 70               STA T1
4880  DB3F  85 71               STA T2
4881  DB41  A5 D6               LDA SSIND
4882  DB43  38                  SEC
4883  DB44  E9 0E               SBC #SSIOFF-2
4884  DB46  85 72               STA T3
4885  DB48  20 51 DF            JSR SSCALC
4886  DB4B  A6 82               LDX LINDX
4887  DB4D  A5 70               LDA T1
4888  DB4F  95 B5               STA NBKL,X
4889  DB51  A5 71               LDA T2
4890  DB53  95 BB               STA NBKH,X
4891  DB55  A9 40               LDA #DYFILE
4892  DB57  20 A6 DD            JSR TSTFLG
4893  DB5A  F0 03               BEQ CLSR1
4894  DB5C  20 A5 DB            JSR CLSDIR
4895  DB5F  4C 27 D2     CLSR1  JMP FRECHN
4896  DB62               ;
4897  DB62               ; CLOSE A WRITE CHANL
4898  DB62               ;
4899  DB62               CLSWRT                  ;CLOSE SEQ WRITE FILE
4900  DB62  A6 82               LDX LINDX
4901  DB64  B5 B5               LDA NBKL,X
4902  DB66  15 BB               ORA NBKH,X
4903  DB68  D0 0C               BNE CLSW10      ;AT LEAST 1 BLOCK WRITTEN
4904  DB6A               ;
4905  DB6A  20 E8 D4            JSR GETPNT
4906  DB6D  C9 02               CMP #2
4907  DB6F  D0 05               BNE CLSW10      ;AT LEAST 1 BYTE WRITTEN
4908  DB71               ;
4909  DB71  A9 0D               LDA #CR
4910  DB73  20 F1 CF            JSR PUTBYT
4911  DB76               CLSW10
4912  DB76  20 E8 D4            JSR GETPNT
4913  DB79  C9 02               CMP #2
4914  DB7B  D0 0F               BNE CLSW20      ;NOT MT BUFFER
4915  DB7D               ;
```

```
LINE# LOC   CODE         LINE


4916  DB7D  20 1E CF            JSR DBLBUF     ;SWITCH BUFS
4917  DB80               ;
4918  DB80  A6 82               LDX LINDX
4919  DB82  B5 B5               LDA NBKL,X
4920  DB84  D0 02               BNE CLSW15
4921  DB86  D6 BB               DEC NBKH,X
4922  DB88               CLSW15
4923  DB88  D6 B5               DEC NBKL,X
4924  DB8A               ;
4925  DB8A  A9 00               LDA #0
4926  DB8C               CLSW20
4927  DB8C  38                  SEC
4928  DB8D  E9 01               SBC #1         ;BACK UP 1
4929  DB8F  48                  PHA            ;SAVE IT
4930  DB90  A9 00               LDA #0
4931  DB92  20 C8 D4            JSR SETPNT
4932  DB95  20 F1 CF            JSR PUTBYT     ;TLINK=0
4933  DB98  68                  PLA            ;LSTCHR COUNT
4934  DB99  20 F1 CF            JSR PUTBYT
4935  DB9C               ;
4936  DB9C  20 C7 D0            JSR WRTBUF     ;WRITE OUT LAST BUFFER
4937  DB9F  20 99 D5            JSR WATJOB     ;FINISH JOB UP
4938  DBA2  4C 1E CF            JMP DBLBUF     ;MAKE SURE BOTH BUFS OK
4939  DBA5               ;RTS
4940  DBA5               ;
4941  DBA5               ;DIRECTORY CLOSE ON OPEN WRITE FILE
4942  DBA5               ;
4943  DBA5  A6 82        CLSDIR LDX LINDX       ;SAVE LINDX
4944  DBA7  8E 70 02            STX WLINDX     ;&SA
4945  DBAA  A5 83               LDA SA
4946  DBAC  48                  PHA
4947  DBAD  BD 60 02            LDA DSEC,X     ;GET DIRECTORY SECTOR
4948  DBB0  85 81               STA SECTOR
4949  DBB2  BD 66 02            LDA DIND,X     ;GET SECTOR OFFSET
4950  DBB5  8D 94 02            STA INDEX
4951  DBB8  B5 EC               LDA FILTYP,X   ;DRV # IN FILTYP
4952  DBBA  29 01               AND #1
4953  DBBC  85 7F               STA DRVNUM
4954  DBBE  AD D4 FE            LDA DIRTRK
4955  DBC1  85 80               STA TRACK
4956  DBC3  20 93 DF            JSR GETACT     ;ALLOCATE A BUFFER
4957  DBC6  48                  PHA
4958  DBC7  85 F9               STA JOBNUM
4959  DBC9  20 60 D4            JSR DRTRD      ;READ DIRECTORY SECTOR
4960  DBCC  A0 00               LDY #0
4961  DBCE  BD 2F FF            LDA BUFIND,X   ;.X IS JOB#
4962  DBD1  85 87               STA R0+1
4963  DBD3  AD 94 02            LDA INDEX
4964  DBD6  85 86               STA R0
4965  DBD8  B1 86               LDA (R0),Y
4966  DBDA  29 20               AND #$20
4967  DBDC  F0 43               BEQ CLSD5
4968  DBDE  20 25 D1            JSR TYPFIL
4969  DBE1  C9 04               CMP #RELTYP
4970  DBE3  F0 44               BEQ CLSD6
```

```
LINE# LOC   CODE        LINE


4971  DBE5              ;
4972  DBE5  B1 86                LDA (R0),Y
4973  DBE7  29 8F                AND #$8F        ;REPLACE FILE
4974  DBE9  91 86                STA (R0),Y
4975  DBEB  C8                   INY
4976  DBEC  B1 86                LDA (R0),Y
4977  DBEE  85 80                STA TRACK
4978  DBF0  84 71                STY TEMP+2
4979  DBF2  A0 1B                LDY #27         ;EXTRACT REPLACEMENT LINK
4980  DBF4  B1 86                LDA (R0),Y      ; TO LAST SECTOR
4981  DBF6  48                   PHA
4982  DBF7  88                   DEY
4983  DBF8  B1 86                LDA (R0),Y
4984  DBFA  D0 0A                BNE CLSD4
4985  DBFC  85 80                STA TRACK
4986  DBFE  68                   PLA
4987  DBFF  85 81                STA SECTOR
4988  DC01  A9 67                LDA #$67
4989  DC03  20 45 E6             JSR CMDER2
4990  DC06              CLSD4
4991  DC06  48                   PHA
4992  DC07  A9 00                LDA #0
4993  DC09  91 86                STA (R0),Y
4994  DC0B  C8                   INY
4995  DC0C  91 86                STA (R0),Y
4996  DC0E  68                   PLA
4997  DC0F  A4 71                LDY TEMP+2
4998  DC11  91 86                STA (R0),Y
4999  DC13  C8                   INY
5000  DC14  B1 86                LDA (R0),Y
5001  DC16  85 81                STA SECTOR
5002  DC18  68                   PLA
5003  DC19  91 86                STA (R0),Y
5004  DC1B  20 7D C8             JSR DELFIL      ;DELETE OLD FILE
5005  DC1E  4C 29 DC             JMP CLSD6       ;SET CLOSE BIT
5006  DC21              CLSD5
5007  DC21  B1 86                LDA (R0),Y
5008  DC23  29 0F                AND #$F
5009  DC25  09 80                ORA #$80
5010  DC27  91 86                STA (R0),Y
5011  DC29  AE 70 02    CLSD6    LDX WLINDX
5012  DC2C  A0 1C                LDY #28         ;SET # OF BLOCKS
5013  DC2E  B5 B5                LDA NBKL,X
5014  DC30  91 86                STA (R0),Y
5015  DC32  C8                   INY
5016  DC33  B5 BB                LDA NBKH,X
5017  DC35  91 86                STA (R0),Y
5018  DC37  68                   PLA
5019  DC38  AA                   TAX
5020  DC39  A9 90                LDA #WRITE      ;WRITE DIRECTORY SECTOR
5021  DC3B  05 7F                ORA DRVNUM
5022  DC3D  20 90 D5             JSR DOIT
5023  DC40  68                   PLA
5024  DC41  85 83                STA SA
5025  DC43  4C 07 D1             JMP FNDWCH      ;RESTORE LINDX
```

```
LINE# LOC   CODE        LINE


5026  DC46              ;
5027  DC46              ;
5028  DC46              ;.END
5028  DC46              ;
5029  DC46                     .LIB OPNCHNL
```

```
DC46              ; OPCHNL
DC46              ;
DC46              ; OPEN A READ CHANL WITH 2 BUFFERS
DC46              ; WILL INSERT SA IN LINTAB
DC46              ; AND INITS ALL POINTERS.
DC46              ; RELATIVE SS AND PTRS ARE SET.
DC46              ;
DC46  A9 01       OPNRCH LDA #1          ; GET ONE DATA BUFFER
DC48  20 E2 D1           JSR GETRCH
DC4B  20 B6 DC           JSR INITP      ; CLEAR POINTERS
DC4E  AD 4A 02           LDA TYPE
DC51  48                 PHA
DC52  0A                 ASL A
DC53  05 7F              ORA DRVNUM
DC55  95 EC              STA FILTYP,X   ; SET FILE TYPE
DC57  20 9B D0           JSR STRRD      ; READ 1ST ONE OR TWO BLOCKS
DC5A  A6 82              LDX LINDX
DC5C  A5 80              LDA TRACK
DC5E  D0 05              BNE OR10
DC60              ;
DC60  A5 81              LDA SECTOR
DC62  9D 44 02           STA LSTCHR,X   ; SET LAST CHAR PTR
DC65              OR10
DC65  68                 PLA
DC66  C9 04              CMP #RELTYP
DC68  D0 3F              BNE OR30       ; MUST BE SEQUENTIAL STUFF
DC6A              ;
DC6A  A4 83              LDY SA
DC6C  B9 2B 02           LDA LINTAB,Y   ; SET CHANNEL AS R/W
DC6F  09 40              ORA #$40
DC71  99 2B 02           STA LINTAB,Y
DC74              ;
DC74              ;
DC74  AD 58 02           LDA REC
DC77  95 C7              STA RS,X       ; SET RECORD SIZE
DC79              ;
DC79  20 8E D2           JSR GETBUF     ; GET SS BUFFER
DC7C  10 03              BPL OR20
DC7E  4C 0F D2           JMP GBERR      ; NO BUFFER
DC81              OR20
DC81  A6 82              LDX LINDX
DC83  95 CD              STA SS,X
DC85  AC 59 02           LDY TRKSS      ; SET SS TRACK
DC88  84 80              STY TRACK
DC8A  AC 5A 02           LDY SECSS      ; SET SS SECTOR
DC8D  84 81              STY SECTOR
DC8F  20 D3 D6           JSR SETH       ; SET SS HEADER
DC92  20 73 DE           JSR RDSS       ; READ IT IN
DC95  20 99 D5           JSR WATJOB
DC98              OROW
DC98              ;
DC98  A6 82              LDX LINDX
DC9A  A9 02              LDA #2
DC9C  95 C1              STA NR,X       ; SET FOR NXTREC
DC9E              ;
```

```
LINE# LOC   CODE        LINE


5086  DC9E  A9 00              LDA #0
5087  DCA0  20 C8 D4           JSR SETPNT     ; SET FIRST DATA BYTE
5088  DCA3              ;
5089  DCA3  20 53 E1           JSR RD40       ; SET UP 1ST RECORD
5090  DCA6  4C 3E DE           JMP GETHDR     ; RESTORE T&S
5091  DCA9              ;
5092  DCA9              OR30
5093  DCA9  20 56 D1           JSR RDBYT      ; SEQUENTIAL SET UP
5094  DCAC  A6 82              LDX LINDX
5095  DCAE  9D 3E 02           STA CHNDAT,X
5096  DCB1  A9 88              LDA #RDYTLK
5097  DCB3  95 F2              STA CHNRDY,X
5098  DCB5  60                 RTS
5099  DCB6              ;
5100  DCB6              ; INITIALIZE VARIABLES FOR OPEN CHANL
5101  DCB6              ; LSTJOB,SETS ACTIVE BUFFER#,LSTCHR,
5102  DCB6              ; BUFFER POINTERS IN BUFTAB=2
5103  DCB6              ;
5104  DCB6  A6 82       INITP  LDX LINDX
5105  DCB8  B5 A7              LDA BUF0,X
5106  DCBA  0A                 ASL A
5107  DCBB  A8                 TAY
5108  DCBC  A9 02              LDA #2
5109  DCBE  99 99 00           STA BUFTAB,Y
5110  DCC1  B5 AE              LDA BUF1,X
5111  DCC3  09 80              ORA #$80
5112  DCC5  95 AE              STA BUF1,X
5113  DCC7  0A                 ASL A
5114  DCC8  A8                 TAY
5115  DCC9  A9 02              LDA #2
5116  DCCB  99 99 00           STA BUFTAB,Y
5117  DCCE              ;
5118  DCCE  A9 00              LDA #0
5119  DCD0  95 B5              STA NBKL,X
5120  DCD2  95 BB              STA NBKH,X
5121  DCD4  A9 00              LDA #0
5122  DCD6  9D 44 02           STA LSTCHR,X
5123  DCD9  60                 RTS
5124  DCDA              ;
5125  DCDA              ;
5126  DCDA              ;
5127  DCDA              ; OPEN A WRITE CHANL WITH 2 BUFFERS
5128  DCDA  20 F7 F1    OPNWCH JSR INTTS       ; GET FIRST TRACK,SECTOR
5129  DCDD  A9 01              LDA #1
5130  DCDF  20 DF D1           JSR GETWCH     ; GET 1 BUFFERS FOR WRITING
5131  DCE2  20 D0 D6           JSR SETHDR     ; SET UP BUFFER HEADERS
5132  DCE5  20 B6 DC           JSR INITP      ; ZROPNT
5133  DCE8  A6 82              LDX LINDX
5134  DCEA  AD 4A 02           LDA TYPE
5135  DCED  48                 PHA
5136  DCEE  0A                 ASL A
5137  DCEF  05 7F              ORA DRVNUM
5138  DCF1  95 EC              STA FILTYP,X   ; SET FILTYP=SEQ
5139  DCF3  68                 PLA
5140  DCF4  C9 04              CMP #RELTYP
```

```
LINE# LOC   CODE        LINE


5141  DCF6  F0 05              BEQ OW10
5142  DCF8  A9 01              LDA #RDYLST    ; ACTIVE LISTENER
5143  DCFA  95 F2              STA CHNRDY,X
5144  DCFC  60                 RTS
5145  DCFD              ;
5146  DCFD              OW10
5147  DCFD  A4 83              LDY SA
5148  DCFF  B9 2B 02           LDA LINTAB,Y
5149  DD02  29 3F              AND #$3F
5150  DD04  09 40              ORA #$40
5151  DD06  99 2B 02           STA LINTAB,Y   ; SET CHANNEL AS R/W
5152  DD09              ;
5153  DD09  AD 58 02           LDA REC
5154  DD0C  95 C7              STA RS,X       ; SET RECORD SIZE
5155  DD0E              ;
5156  DD0E  20 8E D2           JSR GETBUF     ; GET SS BUFFER
5157  DD11  10 03              BPL OW20
5158  DD13  4C 0F D2           JMP GBERR      ; NO BUFFER
5159  DD16              OW20
5160  DD16  A6 82              LDX LINDX
5161  DD18  95 CD              STA SS,X
5162  DD1A  20 C1 DE           JSR CLRBUF
5163  DD1D              ;
5164  DD1D  20 6C F1           JSR NXTTS
5165  DD20  A5 80              LDA TRACK
5166  DD22  8D 59 02           STA TRKSS      ; SAVE SS T&S
5167  DD25  A5 81              LDA SECTOR
5168  DD27  8D 5A 02           STA SECSS
5169  DD2A              ;
5170  DD2A  A6 82              LDX LINDX
5171  DD2C  B5 CD              LDA SS,X
5172  DD2E  20 D3 D6           JSR SETH       ; SET SS HEADER
5173  DD31  A9 00              LDA #0
5174  DD33  20 E9 DE           JSR SETSSP
5175  DD36  A9 00              LDA #0         ; SET NULL LINK
5176  DD38  20 8D DD           JSR PUTSS
5177  DD3B  A9 11              LDA #SSIOFF+1  ; SET LAST CHAR
5178  DD3D  20 8D DD           JSR PUTSS
5179  DD40  A9 00              LDA #0         ; SET THIS SS #
5180  DD42  20 8D DD           JSR PUTSS
5181  DD45  AD 58 02           LDA REC        ; RECORD SIZE
5182  DD48  20 8D DD           JSR PUTSS
5183  DD4B  A5 80              LDA TRACK
5184  DD4D  20 8D DD           JSR PUTSS
5185  DD50  A5 81              LDA SECTOR
5186  DD52  20 8D DD           JSR PUTSS
5187  DD55  A9 10              LDA #SSIOFF
5188  DD57  20 E9 DE           JSR SETSSP
5189  DD5A  20 3E DE           JSR GETHDR     ; GET FIRST T&S
5190  DD5D  A5 80              LDA TRACK
5191  DD5F  20 8D DD           JSR PUTSS
5192  DD62  A5 81              LDA SECTOR
5193  DD64  20 8D DD           JSR PUTSS
5194  DD67              ;
5195  DD67  20 6C DE           JSR WRTSS      ; WRITE IT OUT
```

```
LINE# LOC   CODE         LINE


5196  DD6A  20 99 D5            JSR WATJOB
5197  DD6D  A9 02               LDA #2
5198  DD6F  20 C8 D4            JSR SETPNT
5199  DD72               ;
5200  DD72  A6 82               LDX LINDX      ; SET NR FOR NULL BUFFER
5201  DD74  38                  SEC
5202  DD75  A9 00               LDA #0
5203  DD77  F5 C7               SBC RS,X
5204  DD79  95 C1               STA NR,X
5205  DD7B               ;
5206  DD7B  20 E2 E2            JSR NULBUF     ; NULL RECORDS
5207  DD7E  20 19 DE            JSR NULLNK
5208  DD81  20 5E DE            JSR WRTOUT
5209  DD84  20 99 D5            JSR WATJOB
5210  DD87  20 42 EF            JSR MAPOUT
5211  DD8A  4C 98 DC            JMP OROW
5212  DD8D               ;
5213  DD8D               ;*
5214  DD8D               ;*
5215  DD8D               ;***********************
5216  DD8D               ;*
5217  DD8D               ;* PUTSS
5218  DD8D               ;*
5219  DD8D               ;* PUT BYTE INTO SIDE SECTOR
5220  DD8D               ;*
5221  DD8D               ;***********************
5222  DD8D               ;*
5223  DD8D               ;*
5224  DD8D  48           PUTSS  PHA
5225  DD8E  A6 82               LDX LINDX
5226  DD90  B5 CD               LDA SS,X
5227  DD92  4C FD CF            JMP PUTB1
5228  DD95               ;
5229  DD95               ;.END
5229  DD95               ;
5230  DD95                      .LIB TSTFLG
```

```
LINE# LOC   CODE        LINE


5232  DD95              ;*
5233  DD95              ;*
5234  DD95              ;**********************************
5235  DD95              ;*
5236  DD95              ;* SCFLG
5237  DD95              ;* SETFLG
5238  DD95              ;* CLRFLG
5239  DD95              ;* TSTFLG
5240  DD95              ;*
5241  DD95              ;*********************************
5242  DD95              ;*
5243  DD95              ;*
5244  DD95  90 06       SCFLG  BCC CLRFLG
5245  DD97              ;
5246  DD97  A6 82       SETFLG LDX LINDX
5247  DD99  15 EC              ORA FILTYP,X
5248  DD9B  D0 06              BNE CLRF10
5249  DD9D  A6 82       CLRFLG LDX LINDX
5250  DD9F  49 FF              EOR #$FF
5251  DDA1  35 EC              AND FILTYP,X
5252  DDA3  95 EC       CLRF10 STA FILTYP,X
5253  DDA5  60                 RTS
5254  DDA6              ;
5255  DDA6              ;
5256  DDA6  A6 82       TSTFLG LDX LINDX
5257  DDA8  35 EC              AND FILTYP,X
5258  DDAA  60                 RTS
5259  DDAB              ;
5260  DDAB              ;*
5261  DDAB              ;*
5262  DDAB              ;******************************
5263  DDAB              ;*
5264  DDAB              ;*
5265  DDAB              ;* TSTWRT
5266  DDAB              ;*
5267  DDAB              ;******************************
5268  DDAB              ;*
5269  DDAB              ;*
5270  DDAB  20 93 DF    TSTWRT JSR GETACT
5271  DDAE  AA                 TAX
5272  DDAF  BD 5B 02           LDA LSTJOB,X
5273  DDB2  29 F0              AND #$F0
5274  DDB4  C9 90              CMP #$90
5275  DDB6  60                 RTS
5276  DDB7              ;*
5277  DDB7              ;*
```

```
LINE# LOC   CODE        LINE


5279  DDB7              ;
5280  DDB7              ; TEST FOR ACTIVE FILES FROM
5281  DDB7              ; LINDX TABLE
5282  DDB7              ;
5283  DDB7              ; C=1 FILE NOT ACTIVE X=18,Y=?,A=?
5284  DDB7              ; C=0 FILE ACTIVE X=ENTFND,Y=LINDX,A=?
5285  DDB7              ;
5286  DDB7              ;
5287  DDB7  A2 00       TSTCHN LDX #0          ; START SEARCH AT TOP
5288  DDB9  86 71       TSTC20 STX TEMP+2      ;SAVE TO LOOK ON
5289  DDBB  BD 2B 02           LDA LINTAB,X   ;GET LINDX
5290  DDBE  C9 FF              CMP #$FF
5291  DDC0  D0 08              BNE TSTC40     ;IF PLUS TEST IT
5292  DDC2  A6 71       TSTC30 LDX TEMP+2      ;NOT ACTIVE
5293  DDC4  E8                 INX
5294  DDC5  E0 10              CPX #MAXSA-2   ;SEARCHED ALL
5295  DDC7  90 F0              BCC TSTC20     ;NO
5296  DDC9  60          TSTRTS RTS             ;YES
5297  DDCA              ;
5298  DDCA              ;
5299  DDCA  86 71       TSTC40 STX TEMP+2      ;SAVE X
5300  DDCC  29 3F              AND #$3F
5301  DDCE  A8                 TAY            ;USE LINDX AS INDEX
5302  DDCF  B9 EC 00           LDA FILTYP,Y   ;RIGHT DRIVE # ?
5303  DDD2  29 01              AND #1
5304  DDD4  85 70              STA TEMP+1
5305  DDD6  AE 53 02           LDX ENTFND     ;INDEX ENTRY FOUND
5306  DDD9  B5 E2              LDA FILDRV,X
5307  DDDB  29 01              AND #1
5308  DDDD  C5 70              CMP TEMP+1     ;SAME DRIVE # ?
5309  DDDF  D0 E1              BNE TSTC30     ;NO
5310  DDE1  B9 60 02           LDA DSEC,Y     ;YES - SAME DIR. ENTRY ?
5311  DDE4  D5 D8              CMP ENTSEC,X
5312  DDE6  D0 DA              BNE TSTC30     ;NO
5313  DDE8  B9 66 02           LDA DIND,Y
5314  DDEB  D5 DD              CMP ENTIND,X
5315  DDED  D0 D3              BNE TSTC30     ;NO
5316  DDEF  18                 CLC            ;SET FLAG
5317  DDF0  60                 RTS
5318  DDF1              ;
5319  DDF1              ;
5320  DDF1              ;
5321  DDF1              ;.END
5321  DDF1              ;
5322  DDF1                     .LIB TSUTIL
```

```
LINE# LOC   CODE        LINE


5324  DDF1              ;*
5325  DDF1              ;*
5326  DDF1              ;*********************************
5327  DDF1              ;*
5328  DDF1              ;* SCRUB
5329  DDF1              ;*
5330  DDF1              ;* WRITE OUT BUFFER IF DIRTY
5331  DDF1              ;*
5332  DDF1              ;*********************************
5333  DDF1              ;*
5334  DDF1              ;*
5335  DDF1              SCRUB
5336  DDF1  20 9E DF           JSR GAFLGS
5337  DDF4  50 06              BVC SCR1       ;NOT DIRTY
5338  DDF6              ;
5339  DDF6  20 5E DE           JSR WRTOUT
5340  DDF9  20 99 D5           JSR WATJOB
5341  DDFC  60          SCR1   RTS
5342  DDFD              ;*
5343  DDFD              ;*
5344  DDFD              ;*********************************
5345  DDFD              ;*
5346  DDFD              ;* SETLNK
5347  DDFD              ;*
5348  DDFD              ;* PUT TRACK,SECTOR INTO BUFFER
5349  DDFD              ;*
5350  DDFD              ;*********************************
5351  DDFD              ;*
5352  DDFD              ;*
5353  DDFD  20 2B DE    SETLNK JSR SET00
5354  DE00              ;
5355  DE00  A5 80              LDA TRACK
5356  DE02  91 94              STA (DIRBUF),Y
5357  DE04  C8                 INY
5358  DE05  A5 81              LDA SECTOR
5359  DE07  91 94              STA (DIRBUF),Y
5360  DE09  4C 05 E1           JMP SDIRTY
5361  DE0C              ;
5362  DE0C              ;*
5363  DE0C              ;*
5364  DE0C              ;********************************
5365  DE0C              ;*
5366  DE0C              ;* GETLNK
5367  DE0C              ;*
5368  DE0C              ;* GET LINK FROM BUFFER INTO
5369  DE0C              ;* TRACK AND SECTOR
5370  DE0C              ;*
5371  DE0C              ;********************************
5372  DE0C              ;*
5373  DE0C              ;*
5374  DE0C  20 2B DE    GETLNK JSR SET00
5375  DE0F              ;
5376  DE0F  B1 94              LDA (DIRBUF),Y
5377  DE11  85 80              STA TRACK
5378  DE13  C8                 INY
```

```
LINE# LOC   CODE        LINE


5379  DE14  B1 94              LDA (DIRBUF),Y
5380  DE16  85 81              STA SECTOR
5381  DE18  60                 RTS
5382  DE19              ;*
5383  DE19              ;*
5384  DE19              ;*********************************
5385  DE19              ;*
5386  DE19              ;* NULLNK
5387  DE19              ;*
5388  DE19              ;* SET TRACK LINK=0 & SECTOR
5389  DE19              ;* LINK=LAST NON-ZERO CHAR.
5390  DE19              ;*
5391  DE19              ;**********************************
5392  DE19              ;*
5393  DE19              ;*
5394  DE19              NULLNK
5395  DE19  20 2B DE           JSR SET00
5396  DE1C  A9 00              LDA #0
5397  DE1E  91 94              STA (DIRBUF),Y
5398  DE20  C8                 INY
5399  DE21  A6 82              LDX LINDX
5400  DE23  B5 C1              LDA NR,X
5401  DE25  AA                 TAX
5402  DE26  CA                 DEX
5403  DE27  8A                 TXA
5404  DE28  91 94              STA (DIRBUF),Y
5405  DE2A  60                 RTS
5406  DE2B              ;
5407  DE2B              ;*
5408  DE2B              ;*
5409  DE2B              ;********************************
5410  DE2B              ;*
5411  DE2B              ;* SET00
5412  DE2B              ;*
5413  DE2B              ;* SETUP UP POINTER TO BUFFER
5414  DE2B              ;*
5415  DE2B              ;********************************
5416  DE2B              ;*
5417  DE2B              ;*
5418  DE2B  20 93 DF    SET00  JSR GETACT
5419  DE2E  0A                 ASL A
5420  DE2F  AA                 TAX
5421  DE30  B5 9A              LDA BUFTAB+1,X
5422  DE32  85 95              STA DIRBUF+1
5423  DE34  A9 00              LDA #0
5424  DE36  85 94              STA DIRBUF
5425  DE38  A0 00              LDY #0
5426  DE3A  60                 RTS
5427  DE3B              ;
5428  DE3B              ;*
5429  DE3B              ;*
5430  DE3B              ;********************************
5431  DE3B              ;*
5432  DE3B              ;* GETHDR
5433  DE3B              ;*
```

```
LINE# LOC   CODE        LINE


5434  DE3B              ;* READ TRACK,SETOR FROM HEADER
5435  DE3B              ;*
5436  DE3B              ;*******************************
5437  DE3B              ;*
5438  DE3B              ;*
5439  DE3B  20 EB D0    CURBLK JSR FNDRCH
5440  DE3E  20 93 DF    GETHDR JSR GETACT
5441  DE41  85 F9              STA JOBNUM
5442  DE43  0A                 ASL A
5443  DE44  A8                 TAY
5444  DE45  B9 06 00           LDA HDRS,Y      ; 4/12**************
5445  DE48  85 80              STA TRACK
5446  DE4A  B9 07 00           LDA HDRS+1,Y    ; 4/12**************
5447  DE4D  85 81              STA SECTOR
5448  DE4F  60                 RTS
5449  DE50              ;
5450  DE50              ;*
5451  DE50              ;*
5452  DE50              ;******************************
5453  DE50              ;*
5454  DE50              ;* WRTAB,RDAB WRTOUT,RDIN
5455  DE50              ;* WRTSS,RDSS
5456  DE50              ;*
5457  DE50              ;******************************
5458  DE50              ;*
5459  DE50              ;*
5460  DE50  A9 90       WRTAB  LDA #WRITE
5461  DE52  8D 4D 02           STA CMD
5462  DE55  D0 28              BNE SJ10
5463  DE57              ;
5464  DE57  A9 80       RDAB   LDA #READ
5465  DE59  8D 4D 02           STA CMD
5466  DE5C  D0 21              BNE SJ10
5467  DE5E              ;
5468  DE5E  A9 90       WRTOUT LDA #WRITE
5469  DE60  8D 4D 02           STA CMD
5470  DE63  D0 26              BNE SJ20
5471  DE65              ;
5472  DE65  A9 80       RDIN   LDA #READ
5473  DE67  8D 4D 02           STA CMD
5474  DE6A  D0 1F              BNE SJ20
5475  DE6C              ;
5476  DE6C  A9 90       WRTSS  LDA #WRITE
5477  DE6E  8D 4D 02           STA CMD
5478  DE71  D0 02              BNE RDS5
5479  DE73              ;
5480  DE73  A9 80       RDSS   LDA #READ
5481  DE75  8D 4D 02    RDS5   STA CMD
5482  DE78  A6 82              LDX LINDX
5483  DE7A  B5 CD              LDA SS,X
5484  DE7C  AA                 TAX
5485  DE7D  10 13              BPL SJ30       ;WAS...BNE SJ30
5486  DE7F              ;
5487  DE7F  20 D0 D6    SJ10   JSR SETHDR
5488  DE82  20 93 DF           JSR GETACT
```

```
LINE# LOC   CODE        LINE


5489  DE85  AA                 TAX
5490  DE86  A5 7F              LDA DRVNUM
5491  DE88  9D 5B 02           STA LSTJOB,X
5492  DE8B  20 15 E1    SJ20   JSR CDIRTY
5493  DE8E  20 93 DF           JSR GETACT
5494  DE91  AA                 TAX
5495  DE92  4C 06 D5    SJ30   JMP SETLJB
5496  DE95              ;*
5497  DE95              ;*
5498  DE95              ;*
5499  DE95              ;***************************
5500  DE95              ;*
5501  DE95              ;* RDLNK
5502  DE95              ;*
5503  DE95              ;***************************
5504  DE95              ;*
5505  DE95              ;*
5506  DE95  A9 00       RDLNK  LDA #0
5507  DE97  20 C8 D4           JSR SETPNT
5508  DE9A  20 37 D1           JSR GETBYT
5509  DE9D  85 80              STA TRACK
5510  DE9F  20 37 D1           JSR GETBYT
5511  DEA2  85 81              STA SECTOR
5512  DEA4  60                 RTS
5513  DEA5              ;
5514  DEA5              ;.END
5514  DEA5              ;
5515  DEA5                     .LIB SSUTIL
```

```
LINE# LOC   CODE        LINE


5517  DEA5              ;*********************************
5518  DEA5              ;* B0TOB0: TRANSFER BYTES FROM   *
5519  DEA5              ;* ONE BUF TO OTHER.             *
5520  DEA5              ;* REG: IN: .A= # BYTES          *
5521  DEA5              ;* .Y= SOURCE BUF #              *
5522  DEA5              ;* .X= DESTIN BUF #              *
5523  DEA5              ;*********************************
5524  DEA5              ;
5525  DEA5              B0TOB0
5526  DEA5  48                 PHA
5527  DEA6  A9 00              LDA #0
5528  DEA8  85 6F              STA TEMP
5529  DEAA  85 71              STA TEMP+2
5530  DEAC  B9 2F FF           LDA BUFIND,Y
5531  DEAF  85 70              STA TEMP+1
5532  DEB1  BD 2F FF           LDA BUFIND,X
5533  DEB4  85 72              STA TEMP+3
5534  DEB6  68                 PLA
5535  DEB7  A8                 TAY
5536  DEB8  88                 DEY
5537  DEB9              B02
5538  DEB9  B1 6F              LDA (TEMP),Y
5539  DEBB  91 71              STA (TEMP+2),Y
5540  DEBD  88                 DEY
5541  DEBE  10 F9              BPL B02
5542  DEC0  60                 RTS
5543  DEC1              ;
5544  DEC1              ;*********************************
5545  DEC1              ;* CLRBUF: CLEAR BUFFER GIVEN    *
5546  DEC1              ;* REG: IN: .A= BUFFER #         *
5547  DEC1              ;* OUT: .Y,.A =0                 *
5548  DEC1              ;*********************************
5549  DEC1              ;
5550  DEC1              CLRBUF
5551  DEC1  A8                 TAY
5552  DEC2  B9 2F FF           LDA BUFIND,Y
5553  DEC5  85 70              STA TEMP+1
5554  DEC7  A9 00              LDA #0
5555  DEC9  85 6F              STA TEMP
5556  DECB  A8                 TAY
5557  DECC              CB10
5558  DECC  91 6F              STA (TEMP),Y
5559  DECE  C8                 INY
5560  DECF  D0 FB              BNE CB10
5561  DED1  60                 RTS
5562  DED2              ;
5563  DED2              ;
5564  DED2              ;*********************************
5565  DED2              ;* SSSET: SET SS PNTR TO 0       *
5566  DED2              ;* REG: OUT: .A= SS NUMBER       *
5567  DED2              ;*********************************
5568  DED2              ;
5569  DED2              SSSET
5570  DED2  A9 00              LDA #0
5571  DED4  20 DC DE           JSR SSDIR
```

```
LINE# LOC   CODE        LINE


5572  DED7  A0 02              LDY #2
5573  DED9  B1 94              LDA (DIRBUF),Y
5574  DEDB  60                 RTS
5575  DEDC              ;
5576  DEDC              ;*********************************
5577  DEDC              ;* SSDIR: SET DIRBUF WITH CURRENT*
5578  DEDC              ;* SS POINTER.                   *
5579  DEDC              ;* REGS: IN: .A= LOW BYTE        *
5580  DEDC              ;*********************************
5581  DEDC              ;
5582  DEDC              SSDIR
5583  DEDC  85 94              STA DIRBUF
5584  DEDE  A6 82              LDX LINDX
5585  DEE0  B5 CD              LDA SS,X
5586  DEE2  AA                 TAX
5587  DEE3  BD 2F FF           LDA BUFIND,X
5588  DEE6  85 95              STA DIRBUF+1
5589  DEE8  60                 RTS
5590  DEE9              ;
5591  DEE9              ;*********************************
5592  DEE9              ;* SETSSP: SET DIRBUF & BUFTAB   *
5593  DEE9              ;* WITH CURRENT SS PTR.          *
5594  DEE9              ;* REGS: IN: .A= LOW BYTE        *
5595  DEE9              ;*********************************
5596  DEE9              ;
5597  DEE9              SETSSP
5598  DEE9  48                 PHA
5599  DEEA  20 DC DE           JSR SSDIR
5600  DEED  48                 PHA
5601  DEEE  8A                 TXA
5602  DEEF  0A                 ASL A
5603  DEF0  AA                 TAX
5604  DEF1  68                 PLA
5605  DEF2  95 9A              STA BUFTAB+1,X
5606  DEF4  68                 PLA
5607  DEF5  95 99              STA BUFTAB,X
5608  DEF7  60                 RTS
5609  DEF8              ;
5610  DEF8              ;*********************************
5611  DEF8              ;* SSPOS: POSITION SS & BUFTAB   *
5612  DEF8              ;* TO SSNUM SSIND.               *
5613  DEF8              ;* FLAG: .V=0: OK                *
5614  DEF8              ;* .V=1: OUT OF RANGE            *
5615  DEF8              ;*********************************
5616  DEF8              ;
5617  DEF8              SSPOS
5618  DEF8  20 66 DF           JSR SSTST
5619  DEFB  30 0E              BMI SSP10       ; OUT OF RANGE
5620  DEFD  50 13              BVC SSP20       ; ER0:OK, IN RANGE
5621  DEFF              ;
5622  DEFF  A6 82              LDX LINDX       ; ER1: POSSIBLY IN RANGE
5623  DF01  B5 CD              LDA SS,X
5624  DF03  20 1B DF           JSR IBRD        ; READ SS IN
5625  DF06  20 66 DF           JSR SSTST       ; TEST AGAIN
5626  DF09  10 07              BPL SSP20
```

```
LINE# LOC   CODE        LINE


5627  DF0B              SSP10
5628  DF0B  20 CB E1           JSR SSEND      ; NOT IN RANGE,SET END
5629  DF0E  2C 1D FF           BIT ER1
5630  DF11  60                 RTS
5631  DF12              SSP20
5632  DF12  A5 D6              LDA SSIND      ; OK, SET PTR W/ INDEX
5633  DF14  20 E9 DE           JSR SETSSP
5634  DF17  2C 1C FF           BIT ER0
5635  DF1A  60                 RTS
5636  DF1B              ;
5637  DF1B              ;*********************************
5638  DF1B              ;* IBRD: INDIRECT BLOCK READ &   *
5639  DF1B              ;* IBWT: WRITE.                  *
5640  DF1B              ;* REGS: IN: .A= BUF # FOR R/W   *
5641  DF1B              ;* .X= LINDX                     *
5642  DF1B              ;* (DIRBUF),Y POINTS TO          *
5643  DF1B              ;* T&S TO BE R/W.                *
5644  DF1B              ;*********************************
5645  DF1B              ;
5646  DF1B              IBRD
5647  DF1B  85 F9              STA JOBNUM
5648  DF1D  A9 80              LDA #READ
5649  DF1F  D0 04              BNE IBOP
5650  DF21              IBWT
5651  DF21  85 F9              STA JOBNUM
5652  DF23  A9 90              LDA #WRITE
5653  DF25              IBOP
5654  DF25  48                 PHA
5655  DF26  B5 EC              LDA FILTYP,X
5656  DF28  29 01              AND #1
5657  DF2A  85 7F              STA DRVNUM
5658  DF2C  68                 PLA
5659  DF2D  05 7F              ORA DRVNUM
5660  DF2F  8D 4D 02           STA CMD
5661  DF32              ;
5662  DF32  B1 94              LDA (DIRBUF),Y
5663  DF34  85 80              STA TRACK
5664  DF36  C8                 INY
5665  DF37  B1 94              LDA (DIRBUF),Y
5666  DF39  85 81              STA SECTOR
5667  DF3B  A5 F9              LDA JOBNUM
5668  DF3D  20 D3 D6           JSR SETH
5669  DF40  A6 F9              LDX JOBNUM
5670  DF42  4C 93 D5           JMP DOIT2
5671  DF45              ;
5672  DF45              ;
5673  DF45              ;*
5674  DF45              ;*****************************
5675  DF45              ;*
5676  DF45              ;* GSSPNT
5677  DF45              ;*
5678  DF45              ;*****************************
5679  DF45              ;*
5680  DF45  A6 82       GSSPNT LDX LINDX
5681  DF47  B5 CD              LDA SS,X
```

```
LINE# LOC   CODE        LINE


5682  DF49  4C EB D4           JMP GP1
5683  DF4C              ;
5684  DF4C              SCAL1
5685  DF4C  A9 78              LDA #NSSP
5686  DF4E  20 5C DF           JSR ADDT12     ; ADD (#SS NEEDED)*120
5687  DF51              SSCALC
5688  DF51  CA                 DEX
5689  DF52  10 F8              BPL SCAL1
5690  DF54              ;
5691  DF54  A5 72              LDA T3         ; ADD (# SS INDICES NEEDED)
5692  DF56  4A                 LSR A
5693  DF57  20 5C DF           JSR ADDT12
5694  DF5A  A5 73              LDA T4         ; ADD (# SS BLOCKS NEEDED)
5695  DF5C              ; JMP ADDT12
5696  DF5C              ;
5697  DF5C              ADDT12
5698  DF5C  18                 CLC            ; ADD .A TO T1,T2
5699  DF5D  65 70              ADC T1
5700  DF5F  85 70              STA T1
5701  DF61  90 02              BCC ADDRTS
5702  DF63  E6 71              INC T2
5703  DF65              ADDRTS
5704  DF65  60                 RTS
5705  DF66              ;
5706  DF66  20 D2 DE    SSTST  JSR SSSET
5707  DF69  C5 D5              CMP SSNUM
5708  DF6B  D0 0E              BNE ST20
5709  DF6D  A4 D6              LDY SSIND
5710  DF6F  B1 94              LDA (DIRBUF),Y
5711  DF71  F0 04              BEQ ST10
5712  DF73  2C 1C FF           BIT ER0
5713  DF76  60                 RTS
5714  DF77              ;
5715  DF77              ;.END
5715  DF77              ;
5716  DF77                     .LIB SSTEST
```

```
LINE# LOC   CODE        LINE


5718  DF77              ; FOLLOWING CODE NOT IN 2031
5719  DF77              ; WHICH USES SSTST IN SSUTIL
5720  DF77              ;*********************************
5721  DF77              ;* SSTST: TEST SSNUM & SSIND FOR *
5722  DF77              ;* RESIDENCE & RANGE.            *
5723  DF77              ;* VARS: SSNUM,SSIND,DIRBUF      *
5724  DF77              ;*                               *
5725  DF77              ;* FLAGS: .N RNG .V RES ER       *
5726  DF77              ;* 0 OK 0 YES ER0                *
5727  DF77              ;* 0 MAYBE 1 NO ER1              *
5728  DF77              ;* 1 NO 0 YES ER2                *
5729  DF77              ;* 1 NO 1 NO ER3                 *
5730  DF77              ;*********************************
5731  DF77              ;
5732  DF77              ;SSTEST JSR SSSET       ; SET POINTER & SS #
5733  DF77              ;      CMP SSNUM
5734  DF77              ;      BNE ST20        ; NOT THIS SS
5735  DF77              ;
5736  DF77              ;      LDY SSIND
5737  DF77              ;      LDA (DIRBUF),Y
5738  DF77              ;      BEQ ST10
5739  DF77              ;      BIT ER0         ; OK, RESIDENT
5740  DF77              ;      RTS
5741  DF77              ST10
5742  DF77  2C 1E FF           BIT ER2        ; OUT OF RANGE, RES
5743  DF7A  60                 RTS
5744  DF7B              ST20
5745  DF7B  A5 D5              LDA SSNUM
5746  DF7D  C9 06              CMP #NSSL
5747  DF7F  B0 0A              BCS ST30
5748  DF81  0A                 ASL A
5749  DF82  A8                 TAY
5750  DF83  A9 04              LDA #4
5751  DF85  85 94              STA DIRBUF
5752  DF87  B1 94              LDA (DIRBUF),Y
5753  DF89  D0 04              BNE ST40
5754  DF8B              ST30
5755  DF8B  2C 1F FF           BIT ER3        ; WAY OUT OF RANGE
5756  DF8E  60                 RTS
5757  DF8F              ST40
5758  DF8F  2C 1D FF           BIT ER1        ; NOT RES, RANGE???
5759  DF92  60                 RTS
5760  DF93              ;.END
5760  DF93              ;
5761  DF93                     .LIB GETACT
```

```
LINE# LOC   CODE         LINE


5763  DF93               ;********************************
5764  DF93               ;* GETACT: GET ACTIVE BUFFER #   *
5765  DF93               ;* VARS: BUF0,BUF1,LINDX         *
5766  DF93               ;* REGS: OUT: .A= ACT BUFFER #   *
5767  DF93               ;* .X= LINDX                     *
5768  DF93               ;* FLAGS: .N=1: NO ACT-BUF       *
5769  DF93               ;********************************
5770  DF93               ;
5771  DF93               GETACT
5772  DF93  A6 82               LDX LINDX
5773  DF95  B5 A7               LDA BUF0,X
5774  DF97  10 02               BPL GA1
5775  DF99  B5 AE               LDA BUF1,X
5776  DF9B               GA1
5777  DF9B  29 BF               AND #$BF       ; STRIP DIRTY BIT
5778  DF9D  60                  RTS
5779  DF9E               ;
5780  DF9E               ;********************************
5781  DF9E               ;* GAFLG: GET ACTIVE BUFFER #;   *
5782  DF9E               ;* SET LBUSED & FLAGS.           *
5783  DF9E               ;* REGS: OUT: .A= ACT BUFFER #   *
5784  DF9E               ;* .X= LINDX                     *
5785  DF9E               ;* FLAGS: .N=1: NO ACT-BUF       *
5786  DF9E               ;* .V=1: DIRTY BUF               *
5787  DF9E               ;********************************
5788  DF9E               ;
5789  DF9E               GAFLGS
5790  DF9E  A6 82               LDX LINDX
5791  DFA0  8E 57 02     GA2    STX LBUSED      ;SAVE BUF #
5792  DFA3  B5 A7               LDA BUF0,X
5793  DFA5  10 09               BPL GA3
5794  DFA7               ;
5795  DFA7  8A                  TXA
5796  DFA8  18                  CLC
5797  DFA9  69 07               ADC #MXCHNS+1
5798  DFAB  8D 57 02            STA LBUSED
5799  DFAE  B5 AE               LDA BUF1,X
5800  DFB0               GA3
5801  DFB0  85 70               STA T1
5802  DFB2  29 1F               AND #$1F
5803  DFB4  24 70               BIT T1
5804  DFB6  60                  RTS
5805  DFB7               ;
5806  DFB7               ;*****************************
5807  DFB7               ;*****************************
5808  DFB7               ;
5809  DFB7               ; GET CHANNELS INACTIVE
5810  DFB7               ; BUFFER NUMBER.
5811  DFB7               ;
5812  DFB7               ; INPUT PARAMETERS:
5813  DFB7               ; LINDX - CHANNEL #
5814  DFB7               ;
5815  DFB7               ; OUTPUT PARAMETERS:
5816  DFB7               ; A <== INACTIVE BUFFER #
5817  DFB7               ; OR
```

```
LOC   CODE        LINE


DFB7              ; A <== $FF IF NO
DFB7              ; INACTIVE BUFFER.
DFB7              ;
DFB7              ;*****************************
DFB7              ;
DFB7  A6 82       GETINA LDX LINDX
DFB9  B5 A7              LDA BUF0,X
DFBB  30 02              BMI GI10
DFBD  B5 AE              LDA BUF1,X
DFBF  C9 FF       GI10   CMP #$FF
DFC1  60                 RTS
DFC2              ;
DFC2              ;****************************
DFC2              ;********** P U T I N A ****
DFC2              ;****************************
DFC2              ;
DFC2              ; PUT INACTIVE BUFFER
DFC2              ;
DFC2              ; INPUT PARAMTERS:
DFC2              ; A = BUFFER #
DFC2              ;
DFC2              ; OUTPUT PARAMTERS:
DFC2              ; NONE
DFC2              ;
DFC2              ;****************************
DFC2              ;
DFC2  A6 82       PUTINA LDX LINDX
DFC4  09 80              ORA #$80
DFC6  B4 A7              LDY BUF0,X
DFC8  10 03              BPL PI1
DFCA  95 A7              STA BUF0,X
DFCC  60                 RTS
DFCD  95 AE       PI1    STA BUF1,X
DFCF  60                 RTS
DFD0              ;
DFD0              ; .END
DFD0              ;
DFD0                     .LIB REL1
```

```
LINE# LOC   CODE        LINE


5856  DFD0              ;***********************************
5857  DFD0              ;*
5858  DFD0              ;* ROUTINE: NXTREC
5859  DFD0              ;*
5860  DFD0              ;*
5861  DFD0              ;*
5862  DFD0              ;*
5863  DFD0              ;*
5864  DFD0              ;***********************************
5865  DFD0              NXTREC
5866  DFD0  A9 20              LDA #OVRFLO
5867  DFD2  20 9D DD           JSR CLRFLG
5868  DFD5              ;
5869  DFD5  A9 80              LDA #LRF
5870  DFD7  20 A6 DD           JSR TSTFLG
5871  DFDA  D0 41              BNE NXTR40
5872  DFDC              ;
5873  DFDC  A6 82              LDX LINDX
5874  DFDE  F6 B5              INC RECL,X      ; GOTO NEXT RECORD #
5875  DFE0  D0 02              BNE NXTR15
5876  DFE2  F6 BB              INC RECH,X
5877  DFE4              ;
5878  DFE4              NXTR15
5879  DFE4  A6 82              LDX LINDX
5880  DFE6  B5 C1              LDA NR,X
5881  DFE8  F0 2E              BEQ NXTR45      ; THERE IS A NR
5882  DFEA              ;
5883  DFEA  20 E8 D4           JSR GETPNT      ; GET POINTER
5884  DFED  A6 82              LDX LINDX       ; TEST IF SAME BUFFER
5885  DFEF  D5 C1              CMP NR,X
5886  DFF1  90 03              BCC NXTR20      ; YES, BT<NR
5887  DFF3              ;
5888  DFF3  20 3C E0           JSR NRBUF       ; NO,NEXT BUFFER
5889  DFF6              ;
5890  DFF6  A6 82       NXTR20 LDX LINDX
5891  DFF8  B5 C1              LDA NR,X
5892  DFFA  20 C8 D4           JSR SETPNT      ; ADVANCE TO NEXT REC
5893  DFFD  A1 99              LDA (BUFTAB,X) ; READ 1ST DAT BYTE
5894  DFFF              ;
5895  DFFF  85 85              STA DATA        ; SAVE FOR READ CHANNEL
5896  E001  A9 20              LDA #OVRFLO
5897  E003  20 9D DD           JSR CLRFLG      ; CLEAR
5898  E006              ; THE OVERFLOW FLAG
5899  E006  20 04 E3           JSR ADDNR       ; ADVANCE NR
5900  E009              NXOUT
5901  E009  48                 PHA
5902  E00A  90 28              BCC NXTR30      ; NO BLOCK BOUNDARY
5903  E00C              ;
5904  E00C  A9 00              LDA #0
5905  E00E  20 F6 D4           JSR DRDBYT      ; CHECK TRACK LINK
5906  E011  D0 21              BNE NXTR30      ; NOT LAST BLOCK
5907  E013              ;
5908  E013  68                 PLA
5909  E014  C9 02              CMP #2
5910  E016  F0 12              BEQ NXTR50
```

```
LINE# LOC   CODE        LINE


5911  E018              NXTR45
5912  E018  A9 80              LDA #LRF
5913  E01A  20 97 DD           JSR SETFLG
5914  E01D              NXTR40
5915  E01D  20 2F D1           JSR GETPRE
5916  E020  B5 99              LDA BUFTAB,X
5917  E022  99 44 02           STA LSTCHR,Y
5918  E025  A9 0D              LDA #CR
5919  E027  85 85              STA DATA
5920  E029  60                 RTS
5921  E02A              ;
5922  E02A              NXTR50
5923  E02A  20 35 E0           JSR NXTR35
5924  E02D  A6 82              LDX LINDX
5925  E02F  A9 00              LDA #0
5926  E031  95 C1              STA NR,X
5927  E033  60                 RTS
5928  E034              NXTR30
5929  E034  68                 PLA
5930  E035              NXTR35
5931  E035  A6 82              LDX LINDX
5932  E037              ;*
5933  E037  95 C1              STA NR,X
5934  E039  4C 6E E1           JMP SETLST
5935  E03C              ;*
5936  E03C              ;*
5937  E03C              ;**********************************
5938  E03C              ;*
5939  E03C              ;*
5940  E03C              ;* NRBUF
5941  E03C              ;*
5942  E03C              ;*
5943  E03C              ;********************************
5944  E03C              ;*
5945  E03C              ;*
5946  E03C              NRBUF                  ; READ TRK,SEC LINK
5947  E03C  20 D3 D1           JSR SETDRN
5948  E03F  20 95 DE           JSR RDLNK
5949  E042              ;
5950  E042              ;
5951  E042  20 9E DF           JSR GAFLGS     ; TEST IF DIRTY
5952  E045  50 16              BVC NRBU50     ; CLEAN, DONT WRITE OUT
5953  E047              ;
5954  E047  20 5E DE           JSR WRTOUT     ; DIRTY, WRITE OUT
5955  E04A  20 1E CF           JSR DBLBUF     ; TOGGLE ACTIVE BUFFER
5956  E04D              ;
5957  E04D  A9 02              LDA #2
5958  E04F  20 C8 D4           JSR SETPNT
5959  E052  20 AB DD           JSR TSTWRT     ; TEST IF LSTJOB IS WRT
5960  E055  D0 24              BNE NRBU20     ; NOT A WRITE,BUFFER OK
5961  E057              ;
5962  E057  20 57 DE           JSR RDAB       ; READ IN NEEDED BUFFER
5963  E05A  4C 99 D5           JMP WATJOB     ; WAIT AROUND TILL DONE
5964  E05D              ;
5965  E05D  20 1E CF    NRBU50 JSR DBLBUF      ; TOGGLE ACT BUF
```

```
LINE# LOC   CODE        LINE


5966  E060  20 AB DD           JSR TSTWRT     ; WAS LSTJOB A WRT?
5967  E063  D0 06              BNE NRBU70     ; NOT A WRITE
5968  E065              ;
5969  E065  20 57 DE           JSR RDAB       ; READ IN NEEDED BUFFER
5970  E068  20 99 D5           JSR WATJOB     ; WAIT TILL DONE
5971  E06B              ;
5972  E06B              NRBU70                 ; READ TRK,SEC LINK
5973  E06B  20 95 DE           JSR RDLNK      ; TO DO A READ AHEAD
5974  E06E              ;
5975  E06E  A5 80              LDA TRACK      ; TEST IF LAST BUFFER
5976  E070  F0 09              BEQ NRBU20     ; YES,NO DBL BUFF TODO
5977  E072              ;
5978  E072  20 1E CF           JSR DBLBUF     ; START READ JOB ON THE
5979  E075  20 57 DE           JSR RDAB       ; INACTIVE BUFFER
5980  E078  20 1E CF           JSR DBLBUF
5981  E07B              ;
5982  E07B  60          NRBU20 RTS
5983  E07C              ;
5984  E07C              ;.END
5984  E07C              ;
5985  E07C                     .LIB REL2
```

```
LINE# LOC   CODE         LINE


5987  E07C               ;********************************
5988  E07C               ;*
5989  E07C               ;* RELPUT
5990  E07C               ;*
5991  E07C               ;*
5992  E07C               ;********************************
5993  E07C               ;*
5994  E07C               ;*
5995  E07C  20 05 E1     RELPUT JSR SDIRTY      ; WRITE DATA TO BUFFER
5996  E07F  20 93 DF            JSR GETACT
5997  E082  0A                  ASL A
5998  E083  AA                  TAX
5999  E084  A5 85               LDA DATA
6000  E086  81 99               STA (BUFTAB,X)
6001  E088                   ;
6002  E088  B4 99               LDY BUFTAB,X   ; INC THE POINTER
6003  E08A  C8                  INY
6004  E08B  D0 09               BNE RELP05
6005  E08D  A4 82               LDY LINDX
6006  E08F  B9 C1 00            LDA NR,Y
6007  E092  F0 0A               BEQ RELP07
6008  E094                   ;
6009  E094                   RELP06
6010  E094  A0 02               LDY #2
6011  E096                   RELP05
6012  E096  98                  TYA
6013  E097  A4 82               LDY LINDX
6014  E099                   ;
6015  E099  D9 C1 00            CMP NR,Y       ; TEST IF NR=POINTER
6016  E09C  D0 05               BNE RELP10     ; NO,SET NEW POINTER
6017  E09E                   ;
6018  E09E  A9 20            RELP07 LDA #OVRFLO     ; YES,SET OVERFLOW
6019  E0A0  4C 97 DD            JMP SETFLG
6020  E0A3                   ;
6021  E0A3                   RELP10                 ; WRITE BACK NEW POINTER
6022  E0A3  F6 99               INC BUFTAB,X
6023  E0A5                   ;
6024  E0A5  D0 03               BNE RELP20     ; TEST IF =0
6025  E0A7  20 3C E0            JSR NRBUF      ; PREPARE NXT BUFFER
6026  E0AA                   ;
6027  E0AA  60               RELP20 RTS
6028  E0AB                   ;*
6029  E0AB                   ;*
6030  E0AB                   ;*
6031  E0AB                   ;*******************************
6032  E0AB                   ;*
6033  E0AB                   ;* WRTREL
6034  E0AB                   ;*
6035  E0AB                   ;*
6036  E0AB                   ;*******************************
6037  E0AB                   ;*
6038  E0AB                   ;*
6039  E0AB                   WRTREL
6040  E0AB  A9 A0               LDA #LRF+OVRFLO ; CHECK ALL FLAGS
6041  E0AD  20 A6 DD            JSR TSTFLG
```

```
LINE# LOC   CODE        LINE


6042  E0B0  D0 27              BNE WR50       ; SOME FLAG IS SET
6043  E0B2              WR10
6044  E0B2  A5 85              LDA DATA       ; READY TO PUT DATA
6045  E0B4  20 7C E0           JSR RELPUT
6046  E0B7              WR20
6047  E0B7  A5 F8              LDA EOIFLG
6048  E0B9  F0 0D              BEQ WR40       ; EOI WAS SENT
6049  E0BB  60                 RTS
6050  E0BC              WR30
6051  E0BC  A9 20              LDA #OVRFLO
6052  E0BE  20 A6 DD           JSR TSTFLG
6053  E0C1  F0 05              BEQ WR40       ; NO REC OVERFLOW
6054  E0C3  A9 51              LDA #RECOVF
6055  E0C5  8D 6C 02           STA ERWORD     ; SET ERROR FOR END OF PRINT
6056  E0C8              WR40
6057  E0C8  20 F3 E0           JSR CLREC      ; CLEAR REST OF RECORD
6058  E0CB  20 53 E1           JSR RD40
6059  E0CE  AD 6C 02           LDA ERWORD
6060  E0D1  F0 03              BEQ WR45
6061  E0D3  4C C8 C1           JMP CMDERR
6062  E0D6              WR45
6063  E0D6  4C BC E6           JMP OKERR
6064  E0D9              ;
6065  E0D9              ;
6066  E0D9              WR50
6067  E0D9  29 80              AND #LRF
6068  E0DB  D0 05              BNE WR60       ; LAST REC, ADD
6069  E0DD  A5 F8              LDA EOIFLG
6070  E0DF  F0 DB              BEQ WR30
6071  E0E1              WR51
6072  E0E1  60                 RTS
6073  E0E2              ;
6074  E0E2              WR60
6075  E0E2  A5 85              LDA DATA
6076  E0E4  48                 PHA
6077  E0E5  20 1C E3           JSR ADDREL     ; ADD TO FILE
6078  E0E8  68                 PLA
6079  E0E9  85 85              STA DATA
6080  E0EB  A9 80              LDA #LRF
6081  E0ED  20 9D DD           JSR CLRFLG
6082  E0F0  4C B2 E0           JMP WR10
6083  E0F3              ;*
6084  E0F3              ;*
6085  E0F3              ;*
6086  E0F3              ;********************************
6087  E0F3              ;*
6088  E0F3              ;* CLREC
6089  E0F3              ;*
6090  E0F3              ;*********************************
6091  E0F3              ;
6092  E0F3  A9 20       CLREC  LDA #OVRFLO     ; PUT 0'S INTO REST OF RECORD
6093  E0F5  20 A6 DD           JSR TSTFLG
6094  E0F8  D0 0A              BNE CLR10
6095  E0FA              ;
6096  E0FA  A9 00              LDA #0
```

```
LINE# LOC   CODE        LINE


6097  E0FC  85 85              STA DATA
6098  E0FE  20 7C E0           JSR RELPUT
6099  E101              ;
6100  E101  4C F3 E0           JMP CLREC
6101  E104              ;
6102  E104  60          CLR10  RTS
6103  E105              ;
6104  E105              ;
6105  E105              ;*
6106  E105              ;*
6107  E105              ;******************************
6108  E105              ;*
6109  E105              ;* SDIRTY
6110  E105              ;*
6111  E105              ;******************************
6112  E105              ;*
6113  E105              ;
6114  E105  A9 40       SDIRTY LDA #DYFILE
6115  E107  20 97 DD           JSR SETFLG
6116  E10A  20 9E DF           JSR GAFLGS
6117  E10D  09 40              ORA #$40
6118  E10F  AE 57 02           LDX LBUSED
6119  E112  95 A7              STA BUF0,X
6120  E114  60                 RTS
6121  E115              ;
6122  E115              ;*
6123  E115              ;*
6124  E115              ;******************************
6125  E115              ;*
6126  E115              ;* CDIRTY
6127  E115              ;*
6128  E115              ;******************************
6129  E115              ;*
6130  E115              ;
6131  E115  20 9E DF    CDIRTY JSR GAFLGS
6132  E118  29 BF              AND #$BF
6133  E11A  AE 57 02           LDX LBUSED
6134  E11D  95 A7              STA BUF0,X
6135  E11F  60                 RTS
6136  E120              ;
6137  E120              ;
6138  E120              ;.END
6138  E120              ;
6139  E120                     .LIB REL3
```

```
LINE# LOC   CODE        LINE


6141  E120              ;*******************************
6142  E120              ;*
6143  E120              ;*
6144  E120              ;* RDREL
6145  E120              ;*
6146  E120              ;*
6147  E120              ;*******************************
6148  E120              ;*
6149  E120              ;*
6150  E120              ;
6151  E120              RDREL
6152  E120  A9 80              LDA #LRF
6153  E122  20 A6 DD           JSR TSTFLG
6154  E125  D0 37              BNE RD05        ; NO RECORD ERROR
6155  E127              ;
6156  E127              RD10
6157  E127  20 2F D1           JSR GETPRE
6158  E12A  B5 99              LDA BUFTAB,X
6159  E12C  D9 44 02           CMP LSTCHR,Y
6160  E12F  F0 22              BEQ RD40
6161  E131              ;
6162  E131  F6 99              INC BUFTAB,X
6163  E133  D0 06              BNE RD20
6164  E135              ;
6165  E135  20 3C E0           JSR NRBUF
6166  E138              RD15
6167  E138  20 2F D1           JSR GETPRE
6168  E13B              RD20
6169  E13B  A1 99              LDA (BUFTAB,X)
6170  E13D              RD25
6171  E13D  99 3E 02           STA CHNDAT,Y
6172  E140  A9 89              LDA #RNDRDY
6173  E142  99 F2 00           STA CHNRDY,Y
6174  E145  B5 99              LDA BUFTAB,X
6175  E147  D9 44 02           CMP LSTCHR,Y
6176  E14A  F0 01              BEQ RD30
6177  E14C  60                 RTS
6178  E14D              RD30
6179  E14D  A9 81              LDA #RNDEOI
6180  E14F  99 F2 00           STA CHNRDY,Y
6181  E152  60                 RTS
6182  E153              RD40
6183  E153  20 D0 DF           JSR NXTREC
6184  E156  20 2F D1           JSR GETPRE
6185  E159  A5 85              LDA DATA
6186  E15B  4C 3D E1           JMP RD25
6187  E15E              ;
6188  E15E              RD05
6189  E15E  A6 82              LDX LINDX       ; NO RECORD CHAR SET UP
6190  E160  A9 0D              LDA #CR
6191  E162  9D 3E 02           STA CHNDAT,X
6192  E165  A9 81              LDA #RNDEOI
6193  E167  95 F2              STA CHNRDY,X
6194  E169              ;NO RECORD ERROR
6195  E169  A9 50              LDA #NOREC
```

```
LINE# LOC   CODE        LINE


6196  E16B  20 C8 C1           JSR CMDERR
6197  E16E              ;
6198  E16E              ;.END
6198  E16E              ;
6199  E16E                     .LIB REL4
```

```
LINE# LOC   CODE         LINE


6201  E16E               ;*
6202  E16E               ;*
6203  E16E               ;**********************************
6204  E16E               ;*
6205  E16E               ;*
6206  E16E               ;* SETLST
6207  E16E               ;*
6208  E16E               ;*
6209  E16E               ;**********************************
6210  E16E               ;*
6211  E16E               ;*
6212  E16E  A6 82        SETLST LDX LINDX
6213  E170  B5 C1               LDA NR,X
6214  E172  85 87               STA R1
6215  E174  C6 87               DEC R1
6216  E176  C9 02               CMP #2
6217  E178  D0 04               BNE SETL01
6218  E17A  A9 FF               LDA #$FF
6219  E17C  85 87               STA R1
6220  E17E               SETL01
6221  E17E  B5 C7               LDA RS,X
6222  E180  85 88               STA R2
6223  E182               ;
6224  E182  20 E8 D4            JSR GETPNT
6225  E185  A6 82               LDX LINDX
6226  E187  C5 87               CMP R1
6227  E189  90 19               BCC SETL10
6228  E18B  F0 17               BEQ SETL10
6229  E18D               ;
6230  E18D  20 1E CF            JSR DBLBUF
6231  E190  20 B2 E1            JSR FNDLST
6232  E193  90 08               BCC SETL05
6233  E195               ;
6234  E195  A6 82               LDX LINDX
6235  E197  9D 44 02            STA LSTCHR,X
6236  E19A  4C 1E CF            JMP DBLBUF
6237  E19D               ;
6238  E19D  20 1E CF     SETL05 JSR DBLBUF
6239  E1A0  A9 FF               LDA #$FF
6240  E1A2  85 87               STA R1
6241  E1A4               ;
6242  E1A4  20 B2 E1     SETL10 JSR FNDLST
6243  E1A7  B0 03               BCS SETL40
6244  E1A9               ;
6245  E1A9  20 E8 D4            JSR GETPNT
6246  E1AC               ;
6247  E1AC  A6 82        SETL40 LDX LINDX
6248  E1AE  9D 44 02            STA LSTCHR,X
6249  E1B1  60                  RTS
6250  E1B2               ;*
6251  E1B2               ;*
6252  E1B2               ;*
6253  E1B2               ;*********************************
6254  E1B2               ;*
6255  E1B2               ;*
```

```
LINE# LOC   CODE        LINE


6256  E1B2              ;* FNDLST
6257  E1B2              ;*
6258  E1B2              ;*
6259  E1B2              ;*********************************
6260  E1B2              ;*
6261  E1B2              ;*
6262  E1B2              FNDLST
6263  E1B2  20 2B DE           JSR SET00
6264  E1B5  A4 87              LDY R1          ; OFFSET TO START AT
6265  E1B7              ;
6266  E1B7  B1 94       FNDL10 LDA (DIRBUF),Y
6267  E1B9  D0 0D              BNE FNDL20
6268  E1BB              ;
6269  E1BB  88                 DEY
6270  E1BC  C0 02              CPY #2
6271  E1BE  90 04              BCC FNDL30
6272  E1C0              ;
6273  E1C0  C6 88              DEC R2          ; LIMIT COUNTER
6274  E1C2  D0 F3              BNE FNDL10
6275  E1C4              ;
6276  E1C4  C6 88       FNDL30 DEC R2
6277  E1C6  18                 CLC             ; NOT FOUND HERE
6278  E1C7  60                 RTS
6279  E1C8              ;
6280  E1C8  98          FNDL20 TYA              ; FOUND THE END CHAR
6281  E1C9  38                 SEC
6282  E1CA  60                 RTS
6283  E1CB              ;
6284  E1CB              ;.END
6284  E1CB              ;
6285  E1CB                     .LIB SSEND
```

```
LINE# LOC   CODE        LINE


6287  E1CB              ;*********************************
6288  E1CB              ;* SSEND: POSITION SS & BUFTAB   *
6289  E1CB              ;* TO END OF LAST RECORD.        *
6290  E1CB              ;* VARS:                         *
6291  E1CB              ;*********************************
6292  E1CB              ;
6293  E1CB              SSEND
6294  E1CB  20 D2 DE           JSR SSSET
6295  E1CE  85 D5              STA SSNUM
6296  E1D0  A9 04              LDA #4
6297  E1D2  85 94              STA DIRBUF
6298  E1D4  A0 0A              LDY #SSIOFF-6
6299  E1D6  D0 04              BNE SE20
6300  E1D8              SE10
6301  E1D8  88                 DEY
6302  E1D9  88                 DEY
6303  E1DA  30 26              BMI BREAK
6304  E1DC              SE20
6305  E1DC  B1 94              LDA (DIRBUF),Y ; LOOK FOR LAST SS #
6306  E1DE  F0 F8              BEQ SE10       ; T=0: NOT YET
6307  E1E0  98                 TYA
6308  E1E1  4A                 LSR A
6309  E1E2  C5 D5              CMP SSNUM      ; CHECK SS #
6310  E1E4  F0 09              BEQ SE30       ; THIS IS LAST SS
6311  E1E6              ;
6312  E1E6  85 D5              STA SSNUM
6313  E1E8  A6 82              LDX LINDX
6314  E1EA  B5 CD              LDA SS,X
6315  E1EC  20 1B DF           JSR IBRD       ; READ LAST SS
6316  E1EF              SE30
6317  E1EF  A0 00              LDY #0         ; SET SSIND
6318  E1F1  84 94              STY DIRBUF
6319  E1F3  B1 94              LDA (DIRBUF),Y ; DEBUG
6320  E1F5  D0 0B              BNE BREAK
6321  E1F7  C8                 INY
6322  E1F8  B1 94              LDA (DIRBUF),Y
6323  E1FA  A8                 TAY            ; BACK UP TO TRACK
6324  E1FB  88                 DEY
6325  E1FC  84 D6              STY SSIND
6326  E1FE  98                 TYA
6327  E1FF  4C E9 DE           JMP SETSSP
6328  E202              ;
6329  E202              BREAK
6330  E202  A9 67              LDA #$67
6331  E204  20 45 E6           JSR CMDER2
6332  E207              ;.END
6332  E207              ;
6333  E207                     .LIB RECORD
```

```
LINE# LOC   CODE        LINE


6335  E207              ;********************************
6336  E207              ;* RECORD: POSITION RELATIVE     *
6337  E207              ;* POINTERS TO GIVEN             *
6338  E207              ;* RECORD NUMBER OR TO           *
6339  E207              ;* LAST RECORD IF OUT OF         *
6340  E207              ;* RANGE.                        *
6341  E207              ;********************************
6342  E207              ;
6343  E207              RECORD
6344  E207  20 B3 C2           JSR CMDSET     ; INIT TABLES, PTRS
6345  E20A  AD 01 02           LDA CMDBUF+1
6346  E20D  85 83              STA SA
6347  E20F  20 EB D0           JSR FNDRCH
6348  E212  90 05              BCC R20        ; GOT CHANNEL'S LINDEX
6349  E214              ;
6350  E214  A9 70              LDA #NOCHNL    ; NO VALID CHANNEL
6351  E216  20 C8 C1           JSR CMDERR
6352  E219              R20
6353  E219  A9 A0              LDA #LRF+OVRFLO
6354  E21B  20 9D DD           JSR CLRFLG
6355  E21E  20 25 D1           JSR TYPFIL     ; GET FILE TYPE
6356  E221  F0 05              BEQ R30        ; IT IS RELATIVE FILE
6357  E223              ;
6358  E223  A9 64              LDA #MISTYP    ; WRONG TYPE
6359  E225  20 C8 C1           JSR CMDERR
6360  E228              R30
6361  E228  B5 EC              LDA FILTYP,X
6362  E22A  29 01              AND #1
6363  E22C  85 7F              STA DRVNUM
6364  E22E  AD 02 02           LDA CMDBUF+2
6365  E231  95 B5              STA RECL,X     ; GET RECORD #
6366  E233  AD 03 02           LDA CMDBUF+3
6367  E236  95 BB              STA RECH,X
6368  E238  A6 82              LDX LINDX      ; CLEAR CHNRDY TO RNDRDY
6369  E23A  A9 89              LDA #RNDRDY
6370  E23C  95 F2              STA CHNRDY,X
6371  E23E              ;
6372  E23E  AD 04 02           LDA CMDBUF+4   ; GET OFFSET
6373  E241  F0 10              BEQ R40
6374  E243  38                 SEC
6375  E244  E9 01              SBC #1
6376  E246  F0 0B              BEQ R40
6377  E248  D5 C7              CMP RS,X
6378  E24A  90 07              BCC R35
6379  E24C              ;
6380  E24C  A9 51              LDA #RECOVF
6381  E24E  8D 6C 02           STA ERWORD
6382  E251  A9 00              LDA #0
6383  E253              R35
6384  E253              R40
6385  E253  85 D4              STA RECPTR     ; SET OFFSET
6386  E255  20 0E CE           JSR FNDREL     ; CALC SS STUFF
6387  E258  20 F8 DE           JSR SSPOS      ; SET SS PTRS
6388  E25B  50 08              BVC R50
6389  E25D              ;
```

```
LINE# LOC   CODE        LINE


6390  E25D  A9 80              LDA #LRF       ; BEYOND THE END
6391  E25F  20 97 DD           JSR SETFLG     ; SET LAST REC FLAG
6392  E262  4C 5E E1           JMP RD05
6393  E265              R50
6394  E265  20 75 E2           JSR POSITN     ; POSITION TO RECORD
6395  E268  A9 80              LDA #LRF
6396  E26A  20 A6 DD           JSR TSTFLG
6397  E26D  F0 03              BEQ R60
6398  E26F  4C 5E E1           JMP RD05
6399  E272              R60
6400  E272  4C 94 C1           JMP ENDCMD     ; THAT'S ALL
6401  E275              ;
6402  E275              ;********************************
6403  E275              ;* POSITN: POSITION RELATIVE     *
6404  E275              ;* DATA BLOCK INTO ACTIVE        *
6405  E275              ;* BUFFER & NEXT BLOCK           *
6406  E275              ;* INTO INACTIVE BUFFER.         *
6407  E275              ;********************************
6408  E275              ;
6409  E275              POSITN
6410  E275              ; JSR RESTOR ;RESTORE BUFFER AREAS
6411  E275  20 9C E2           JSR POSBUF     ; POSITION BUFFERS
6412  E278  A5 D7              LDA RELPTR
6413  E27A  20 C8 D4           JSR SETPNT     ; SET PTR FROM FNDREL
6414  E27D              ;
6415  E27D  A6 82              LDX LINDX
6416  E27F  B5 C7              LDA RS,X
6417  E281  38                 SEC            ; CALC THE OFFSET
6418  E282  E5 D4              SBC RECPTR
6419  E284  B0 03              BCS P2
6420  E286  4C 02 E2           JMP BREAK      ; SHOULD NOT BE NEEDED
6421  E289              P2
6422  E289  18                 CLC
6423  E28A  65 D7              ADC RELPTR
6424  E28C  90 03              BCC P30
6425  E28E  69 01              ADC #1
6426  E290  38                 SEC
6427  E291              P30
6428  E291  20 09 E0           JSR NXOUT      ; SET NR
6429  E294  4C 38 E1           JMP RD15
6430  E297  A9 51              LDA #RECOVF
6431  E299  20 C8 C1           JSR CMDERR
6432  E29C              ;* POSBUF: POSITION PROPER DATA *
6433  E29C              ;* BLOCKS INTO BUFFERS *
6434  E29C              ;********************************
6435  E29C              ;
6436  E29C              POSBUF
6437  E29C  A5 94              LDA DIRBUF
6438  E29E  85 89              STA R3
6439  E2A0  A5 95              LDA DIRBUF+1
6440  E2A2  85 8A              STA R4
6441  E2A4  20 D0 E2           JSR BHERE      ; IS BUFFER IN?
6442  E2A7  D0 01              BNE P10        ; YES!
6443  E2A9  60                 RTS
6444  E2AA              P10
```

```
LINE# LOC   CODE        LINE


6445  E2AA  20 F1 DD           JSR SCRUB       ; CLEAN BUFFER
6446  E2AD  20 0C DE           JSR GETLNK
6447  E2B0  A5 80              LDA TRACK
6448  E2B2  F0 0E              BEQ P80
6449  E2B4  20 D3 E2           JSR BHERE2
6450  E2B7  D0 06              BNE P75
6451  E2B9  20 1E CF           JSR DBLBUF
6452  E2BC  4C DA D2           JMP FREIAC
6453  E2BF              P75
6454  E2BF  20 DA D2           JSR FREIAC
6455  E2C2              ;
6456  E2C2              P80
6457  E2C2  A0 00              LDY #0          ; GET PROPER BLOCK
6458  E2C4  B1 89              LDA (R3),Y
6459  E2C6  85 80              STA TRACK
6460  E2C8  C8                 INY
6461  E2C9  B1 89              LDA (R3),Y
6462  E2CB  85 81              STA SECTOR
6463  E2CD  4C AF D0           JMP STRDBL      ; GET NEXT BLOCK, TOO.
6464  E2D0              ;RTS
6465  E2D0              ;
6466  E2D0              BHERE
6467  E2D0  20 3E DE           JSR GETHDR      ; GET THE HEADER
6468  E2D3              BHERE2
6469  E2D3  A0 00              LDY #0
6470  E2D5  B1 89              LDA (R3),Y
6471  E2D7  C5 80              CMP TRACK
6472  E2D9  F0 01              BEQ BH10        ; TEST SECTOR, TOO.
6473  E2DB  60                 RTS
6474  E2DC              BH10
6475  E2DC  C8                 INY
6476  E2DD  B1 89              LDA (R3),Y
6477  E2DF  C5 81              CMP SECTOR      ; SET .Z
6478  E2E1  60                 RTS
6479  E2E2              ;
6480  E2E2              ;.END
6480  E2E2              ;
6481  E2E2                     .LIB NULBUF
```

```
LINE# LOC   CODE        LINE


6483  E2E2              ;*********************************
6484  E2E2              ;* NULBUF: SET NULL RECORDS IN   *
6485  E2E2              ;* ACT-BUF FOR EXTENTION         *
6486  E2E2              ;* VARS: NR,RS,LX,ACT-BUF        *
6487  E2E2              ;* IN: NR= LAST RECORD           *
6488  E2E2              ;* POSITION IN PREVIOUS          *
6489  E2E2              ;* BUFFER.                       *
6490  E2E2              ;* OUT: NR= LAST RECORD          *
6491  E2E2              ;* POSITION IN BUFFER            *
6492  E2E2              ;* FOR NEXT NULBUF OR            *
6493  E2E2              ;* TO SET LSTCHR.                *
6494  E2E2              ;*********************************
6495  E2E2              ;
6496  E2E2              NULBUF
6497  E2E2  20 2B DE           JSR SET00      ;SET INDIRECT PTR
6498  E2E5  A0 02              LDY #2
6499  E2E7  A9 00              LDA #0
6500  E2E9              NB10
6501  E2E9  91 94              STA (DIRBUF),Y ;CLEAR BUFFER
6502  E2EB  C8                 INY
6503  E2EC  D0 FB              BNE NB10
6504  E2EE              ;
6505  E2EE  20 04 E3           JSR ADDNR      ;ADVANCE NR
6506  E2F1              NB20
6507  E2F1  95 C1              STA NR,X
6508  E2F3  A8                 TAY
6509  E2F4  A9 FF              LDA #$FF
6510  E2F6  91 94              STA (DIRBUF),Y ;INIT RECORD W/ CR
6511  E2F8  20 04 E3           JSR ADDNR
6512  E2FB  90 F4              BCC NB20       ;NOT DONE
6513  E2FD              ;
6514  E2FD  D0 04              BNE NB30
6515  E2FF  A9 00              LDA #0
6516  E301  95 C1              STA NR,X
6517  E303              NB30
6518  E303  60                 RTS
6519  E304              ;
6520  E304              ; ADD RS & NR, LEAVE IN ACCUM
6521  E304              ; C=1: CROSS BUFFER BOUNDARY
6522  E304              ;
6523  E304              ADDNR
6524  E304  A6 82              LDX LINDX
6525  E306  B5 C1              LDA NR,X
6526  E308  38                 SEC
6527  E309  F0 0D              BEQ AN05
6528  E30B              ;
6529  E30B  18                 CLC
6530  E30C  75 C7              ADC RS,X
6531  E30E  90 0B              BCC AN10
6532  E310  D0 06              BNE AN05
6533  E312  A9 02              LDA #2
6534  E314  2C 1B FF           BIT ER00
6535  E317  60                 RTS
6536  E318              ;
6537  E318              AN05
```

```
LINE# LOC   CODE        LINE


6538  E318  69 01              ADC #1          ; ADJUST FOR LINK
6539  E31A  38                 SEC
6540  E31B              AN10
6541  E31B  60                 RTS
6542  E31C              ;
6543  E31C              ;
6544  E31C              ; .END
6544  E31C              ;
6545  E31C                     .LIB ADDREL
```

```
LINE# LOC   CODE        LINE


E31C              ;*********************************
E31C              ;* ADDREL: ADD BLOCKS TO RELATIVE*
E31C              ;* FILE.                         *
E31C              ;* VARS:                         *
E31C              ;* REGS:                         *
E31C              ;*                               *
E31C              ;*********************************
E31C              ;
E31C              ADDREL
E31C  20 D3 D1           JSR SETDRN
E31F  20 CB E1           JSR SSEND      ; SET UP END OF FILE
E322  20 9C E2           JSR POSBUF
E325  20 7B CF           JSR DBSET
E328  A5 D6              LDA SSIND
E32A  85 87              STA R1         ; SAVE SS INDEX
E32C  A5 D5              LDA SSNUM
E32E  85 86              STA R0         ; SAVE SS NUMBER
E330  A9 00              LDA #0
E332  85 88              STA R2         ; CLEAR FLAG FOR ONE BLOCK
E334              ;
E334  A9 00              LDA #0         ; CLEAR FOR CALCULATION...
E336  85 D4              STA RECPTR     ; ...TO 1ST BYTE IN RECORD
E338  20 0E CE           JSR FNDREL     ; CALC SS PTRS
E33B              ADDR1                  ; ENTRY FOR REL RECORD FIX
E33B  20 9B EF           JSR NUMFRE     ; CALC AVAILABLE...
E33E              ;
E33E  A4 82              LDY LINDX      ; RECORD SPAN?
E340  B6 C7              LDX RS,Y
E342  CA                 DEX
E343  8A                 TXA
E344  18                 CLC
E345  65 D7              ADC RELPTR
E347  90 0C              BCC AR10       ; NO SPAN
E349              ;
E349  E6 D6              INC SSIND      ; INC SS PTRS & CHECK
E34B  E6 D6              INC SSIND      ; INC SS PTRS & CHECK
E34D  D0 06              BNE AR10
E34F  E6 D5              INC SSNUM
E351  A9 10              LDA #SSIOFF
E353  85 D6              STA SSIND
E355              AR10
E355  A5 87              LDA R1
E357  18                 CLC
E358  69 02              ADC #2
E35A  20 E9 DE           JSR SETSSP
E35D              ;
E35D  A5 D5              LDA SSNUM
E35F  C9 06              CMP #NSSL
E361  90 05              BCC AR25       ; VALID RANGE
E363              ;
E363              AR20
E363  A9 52              LDA #BIGFIL
E365  20 C8 C1           JSR CMDERR     ; TOO MANY SS'S
E368              AR25
E368  A5 D6              LDA SSIND      ; CALC # BLOCKS NEEDED...
```

```
LINE# LOC   CODE        LINE


6602  E36A  38                 SEC             ; ...& CHECK AGAINST AVAIL.

6603  E36B  E5 87              SBC R1
6604  E36D  B0 03              BCS AR30
6605  E36F  E9 0F              SBC #SSIOFF-1
6606  E371  18                 CLC
6607  E372              AR30
6608  E372  85 72              STA T3          ; # SS INDICES
6609  E374  A5 D5              LDA SSNUM
6610  E376  E5 86              SBC R0
6611  E378  85 73              STA T4          ; # SS NEEDED
6612  E37A              ;
6613  E37A  A2 00              LDX #0          ; CLEAR ACCUM.
6614  E37C  86 70              STX T1
6615  E37E  86 71              STX T2
6616  E380  AA                 TAX             ; .X=# SS
6617  E381  20 51 DF           JSR SSCALC      ; CALC # OF BLOCKS NEEDED
6618  E384              ;
6619  E384  A5 71              LDA T2
6620  E386  D0 07              BNE AR35
6621  E388  A6 70              LDX T1
6622  E38A  CA                 DEX
6623  E38B  D0 02              BNE AR35
6624  E38D              ;
6625  E38D  E6 88              INC R2
6626  E38F              AR35
6627  E38F  CD 73 02           CMP NBTEMP+1
6628  E392  90 09              BCC AR40        ; OK!!
6629  E394  D0 CD              BNE AR20
6630  E396  AD 72 02           LDA NBTEMP
6631  E399  C5 70              CMP T1
6632  E39B  90 C6              BCC AR20        ; NOT ENUF BLOCKS
6633  E39D              AR40
6634  E39D  A9 01              LDA #1
6635  E39F  20 F6 D4           JSR DRDBYT      ; LOOK AT SECTOR LINK
6636  E3A2  18                 CLC
6637  E3A3  69 01              ADC #1          ; +1 IS NR
6638  E3A5  A6 82              LDX LINDX
6639  E3A7  95 C1              STA NR,X
6640  E3A9  20 6C F1           JSR NXTTS       ; GET NEXT BLOCK...
6641  E3AC  20 FD DD           JSR SETLNK      ; ...& SET LINK.
6642  E3AF  A5 88              LDA R2
6643  E3B1  D0 15              BNE AR50        ; ADD ONE BLOCK
6644  E3B3              ;
6645  E3B3  20 5E DE           JSR WRTOUT      ; WRITE CURRENT LAST REC
6646  E3B6              AR45
6647  E3B6  20 1E CF           JSR DBLBUF      ; SWITCH BUFS
6648  E3B9  20 D0 D6           JSR SETHDR      ; SET HDR FROM T & S
6649  E3BC  20 6C F1           JSR NXTTS       ; GET ANOTHER
6650  E3BF  20 FD DD           JSR SETLNK      ; SET UP LINK
6651  E3C2  20 E2 E2           JSR NULBUF      ; CLEAN IT OUT
6652  E3C5  4C D4 E3           JMP AR55
6653  E3C8              AR50
6654  E3C8  20 1E CF           JSR DBLBUF      ; SWITCH BUFS
6655  E3CB  20 D0 D6           JSR SETHDR      ; SET HDR FROM T & S
6656  E3CE  20 E2 E2           JSR NULBUF      ; CLEAN BUFFER
```

```
LINE# LOC   CODE        LINE


6657  E3D1  20 19 DE           JSR NULLNK     ; LAST BLOCK =0,LSTCHR
6658  E3D4              AR55
6659  E3D4  20 5E DE           JSR WRTOUT     ; WRITE BUFFER
6660  E3D7  20 0C DE           JSR GETLNK     ; GET T&S FROM LINK
6661  E3DA  A5 80              LDA TRACK
6662  E3DC  48                 PHA            ; SAVE 'EM
6663  E3DD  A5 81              LDA SECTOR
6664  E3DF  48                 PHA
6665  E3E0  20 3E DE           JSR GETHDR     ; NOW GET HDR T&S
6666  E3E3  A5 81              LDA SECTOR
6667  E3E5  48                 PHA            ; SAVE 'EM
6668  E3E6  A5 80              LDA TRACK
6669  E3E8  48                 PHA
6670  E3E9  20 45 DF           JSR GSSPNT     ; CHECK SS PTR
6671  E3EC  AA                 TAX
6672  E3ED  D0 0A              BNE AR60
6673  E3EF              ;
6674  E3EF  20 4E E4           JSR NEWSS      ; NEED ANOTHER SS
6675  E3F2  A9 10              LDA #SSIOFF
6676  E3F4  20 E9 DE           JSR SETSSP     ; .A=BT VAL
6677  E3F7  E6 86              INC R0         ; ADVANCE SS COUNT
6678  E3F9              AR60
6679  E3F9  68                 PLA
6680  E3FA  20 8D DD           JSR PUTSS      ; RECORD T&S...
6681  E3FD  68                 PLA
6682  E3FE  20 8D DD           JSR PUTSS      ; ...IN SS.
6683  E401  68                 PLA            ; GET T&S FROM LINK
6684  E402  85 81              STA SECTOR
6685  E404  68                 PLA
6686  E405  85 80              STA TRACK
6687  E407  F0 0F              BEQ AR65       ; T=0: THAT'S ALL!!
6688  E409              ;
6689  E409  A5 86              LDA R0
6690  E40B  C5 D5              CMP SSNUM
6691  E40D  D0 A7              BNE AR45       ; NOT EVEN DONE YET
6692  E40F              ;
6693  E40F  20 45 DF           JSR GSSPNT
6694  E412  C5 D6              CMP SSIND
6695  E414  90 A0              BCC AR45       ; ALMOST DONE
6696  E416  F0 B0              BEQ AR50       ; ONE MORE BLOCK LEFT
6697  E418              AR65
6698  E418  20 45 DF           JSR GSSPNT
6699  E41B  48                 PHA
6700  E41C  A9 00              LDA #0
6701  E41E  20 DC DE           JSR SSDIR
6702  E421  A9 00              LDA #0
6703  E423  A8                 TAY
6704  E424  91 94              STA (DIRBUF),Y
6705  E426  C8                 INY
6706  E427  68                 PLA
6707  E428  38                 SEC
6708  E429  E9 01              SBC #1
6709  E42B  91 94              STA (DIRBUF),Y
6710  E42D  20 6C DE           JSR WRTSS      ; WRITE SS
6711  E430  20 99 D5           JSR WATJOB
```

```
LINE# LOC   CODE        LINE


6712  E433  20 42 EF           JSR MAPOUT
6713  E436  20 0E CE           JSR FNDREL
6714  E439  20 1E CF           JSR DBLBUF      ; GET BACK TO LEADING BUFFER
6715  E43C  20 F8 DE           JSR SSPOS
6716  E43F  70 03              BVS AR70
6717  E441  4C 75 E2           JMP POSITN
6718  E444              AR70
6719  E444  A9 80              LDA #LRF
6720  E446  20 97 DD           JSR SETFLG
6721  E449  A9 50              LDA #NOREC
6722  E44B  20 C8 C1           JSR CMDERR
6723  E44E              ; .END
6723  E44E              ;
6724  E44E                     .LIB NEWSS
```

```
LINE# LOC   CODE         LINE

E44E              ;*********************************
E44E              ;* NEWSS: GENERATE NEW SS & FIX  *
E44E              ;* OLD SS'S TO REFLECT IT.       *
E44E              ;* VARS:                         *
E44E              ;* REGS:                         *
E44E              ;*                               *
E44E              ;*********************************
E44E              ;
E44E              NEWSS
E44E  20 6C F1           JSR NXTTS      ; GET T&S BASED ON HDR
E451  20 1E CF           JSR DBLBUF     ; USE INACTIVE BUFFER
E454  20 F1 DD           JSR SCRUB
E457  20 93 DF           JSR GETACT
E45A  48                 PHA
E45B  20 C1 DE           JSR CLRBUF
E45E  A6 82              LDX LINDX
E460  B5 CD              LDA SS,X       ; SET REGS FOR TRANSFER
E462  A8                 TAY
E463  68                 PLA
E464  AA                 TAX
E465  A9 10              LDA #SSIOFF    ; # OF CHARS
E467  20 A5 DE           JSR B0TOB0     ; TRANSFER AT BUF(0)
E46A              ;
E46A  A9 00              LDA #0
E46C  20 DC DE           JSR SSDIR
E46F  A0 02              LDY #2
E471  B1 94              LDA (DIRBUF),Y ; GET SS #
E473  48                 PHA
E474  A9 00              LDA #0
E476  20 C8 D4           JSR SETPNT
E479  68                 PLA
E47A  18                 CLC
E47B  69 01              ADC #1
E47D  91 94              STA (DIRBUF),Y ; PUT SS # IN NEW SS
E47F  0A                 ASL A
E480  69 04              ADC #4
E482  85 89              STA R3         ; SAVE POSITION
E484  A8                 TAY
E485  38                 SEC
E486  E9 02              SBC #2
E488  85 8A              STA R4
E48A  A5 80              LDA TRACK
E48C  85 87              STA R1         ; SAVE FOR SS UPDATE
E48E  91 94              STA (DIRBUF),Y ; PUT TRACK IN SS
E490  C8                 INY
E491  A5 81              LDA SECTOR
E493  85 88              STA R2         ; SAVE FOR SS UPDATE
E495  91 94              STA (DIRBUF),Y ; PUT SECTOR IN SS
E497  A0 00              LDY #0
E499  98                 TYA
E49A  91 94              STA (DIRBUF),Y ; NULL LINK
E49C  C8                 INY
E49D  A9 11              LDA #SSIOFF+1  ; PTR TO LAST BYTE
E49F  91 94              STA (DIRBUF),Y
E4A1              ;
```

```
LINE# LOC   CODE        LINE


6781  E4A1  A9 10              LDA #SSIOFF
6782  E4A3  20 C8 D4           JSR SETPNT
6783  E4A6  20 50 DE           JSR WRTAB
6784  E4A9  20 99 D5           JSR WATJOB
6785  E4AC              ;
6786  E4AC              NS20
6787  E4AC  A6 82              LDX LINDX
6788  E4AE  B5 CD              LDA SS,X       ; GET SS BUFFER #
6789  E4B0  48                 PHA
6790  E4B1  20 9E DF           JSR GAFLGS
6791  E4B4  A6 82              LDX LINDX
6792  E4B6  95 CD              STA SS,X       ; SWAP ACT-BUF & SS
6793  E4B8  68                 PLA
6794  E4B9  AE 57 02           LDX LBUSED
6795  E4BC  95 A7              STA BUF0,X
6796  E4BE              ;
6797  E4BE  A9 00              LDA #0
6798  E4C0  20 C8 D4           JSR SETPNT     ; SET LINK TO NEW SS
6799  E4C3  A0 00              LDY #0
6800  E4C5  A5 80              LDA TRACK
6801  E4C7  91 94              STA (DIRBUF),Y
6802  E4C9  C8                 INY
6803  E4CA  A5 81              LDA SECTOR
6804  E4CC  91 94              STA (DIRBUF),Y
6805  E4CE  4C DE E4           JMP NS50
6806  E4D1              NS40
6807  E4D1  20 93 DF           JSR GETACT
6808  E4D4  A6 82              LDX LINDX
6809  E4D6  20 1B DF           JSR IBRD       ; READ NEXT SS
6810  E4D9  A9 00              LDA #0
6811  E4DB  20 C8 D4           JSR SETPNT     ; PTR=0
6812  E4DE              NS50
6813  E4DE  C6 8A              DEC R4
6814  E4E0  C6 8A              DEC R4
6815  E4E2  A4 89              LDY R3         ; GET NEW SS LINK PTR
6816  E4E4  A5 87              LDA R1
6817  E4E6  91 94              STA (DIRBUF),Y ; PUT TRACK IN
6818  E4E8  C8                 INY
6819  E4E9  A5 88              LDA R2
6820  E4EB  91 94              STA (DIRBUF),Y ; PUT SECTOR IN
6821  E4ED              ;
6822  E4ED  20 5E DE           JSR WRTOUT     ; WRITE IT BACK...
6823  E4F0  20 99 D5           JSR WATJOB     ; ...& WAIT
6824  E4F3  A4 8A              LDY R4
6825  E4F5  C0 03              CPY #3
6826  E4F7  B0 D8              BCS NS40       ; MORE SS TO UPDATE!
6827  E4F9              ;
6828  E4F9  4C 1E CF           JMP DBLBUF     ; RESET ACTIVE BUFFER
6829  E4FC              ; .END
6829  E4FC              ;
6830  E4FC                     .LIB ERPROC
```

```
LINE# LOC   CODE        LINE


6832  E4FC              ; ERROR PROCESSING
6833  E4FC              ;
6834  E4FC              ; CONTROLLER ERRORS
6835  E4FC              ; 0 (1) NO ERROR
6836  E4FC              ; 20 (2) CAN'T FIND BLOCK HEADER
6837  E4FC              ; 21 (3) NO SYNCH CHARACTER
6838  E4FC              ; 22 (4) DATA BLOCK NOT PRESENT
6839  E4FC              ; 23 (5) CHECKSUM ERROR IN DATA
6840  E4FC              ; 24 (16) BYTE DECODING ERROR
6841  E4FC              ; 25 (7) WRITE-VERIFY ERROR
6842  E4FC              ; 26 (8) WRITE W/ WRITE PROTECT ON
6843  E4FC              ; 27 (9) CHECKSUM ERROR IN HEADER
6844  E4FC              ; 28 (10) DATA EXTENDS INTO NEXT BLOCK
6845  E4FC              ; 29 (11) DISK I.D. MISMATCH
6846  E4FC              ;
6847  E4FC              ; COMMAND ERRORS
6848  E4FC              ; 30 GENERAL SYNTAX
6849  E4FC              ; 31 INVALID COMMAND
6850  E4FC              ; 32 LONG LINE
6851  E4FC              ; 33 INVALID FILNAME
6852  E4FC              ; 34 NO FILE GIVEN
6853  E4FC              ; 39 COMMAND FILE NOT FOUND
6854  E4FC              ;
6855  E4FC              ; 50 RECORD NOT PRESENT
6856  E4FC              ; 51 OVERFLOW IN RECORD
6857  E4FC              ; 52 FILE TOO LARGE
6858  E4FC              ;
6859  E4FC              ; 60 FILE OPEN FOR WRITE
6860  E4FC              ; 61 FILE NOT OPEN
6861  E4FC              ; 62 FILE NOT FOUND
6862  E4FC              ; 63 FILE EXISTS
6863  E4FC              ; 64 FILE TYPE MISMATCH
6864  E4FC              ; 65 NO BLOCK
6865  E4FC              ; 66 ILLEGAL TRACK OR SECTOR
6866  E4FC              ; 67 ILLEGAL SYSTEM T OR S
6867  E4FC              ;
6868  E4FC              ; 70 NO CHANNELS AVAILABLE
6869  E4FC              ; 71 DIRECTORY ERROR
6870  E4FC              ; 72 DISK FULL
6871  E4FC              ; 73 CBM DOS V2.6 V170
6872  E4FC              ; 74 DRIVE NOT READY
6873  E4FC              ;
6874  E4FC              ; 1 FILES SCRATCHED RESPONSE
6875  E4FC              ;
6876  E4FC              BADSYN =$30
6877  E4FC              BADCMD =$31
6878  E4FC              LONGLN =$32
6879  E4FC              BADFN  =$33
6880  E4FC              NOFILE =$34
6881  E4FC              NOCFIL =$39
6882  E4FC              NOREC  =$50
6883  E4FC              RECOVF =$51
6884  E4FC              BIGFIL =$52
6885  E4FC              FILOPN =$60
6886  E4FC              FILNOP =$61
```

```
LINE# LOC   CODE        LINE


6887  E4FC              FLNTFD =$62
6888  E4FC              FLEXST =$63
6889  E4FC              MISTYP =$64
6890  E4FC              NOBLK  =$65
6891  E4FC              BADTS  =$66
6892  E4FC              SYSTS  =$67
6893  E4FC              NOCHNL =$70
6894  E4FC              DIRERR =$71
6895  E4FC              DSKFUL =$72
6896  E4FC              CBMV2  =$73
6897  E4FC              NODRIV =$74
```

```
LINE# LOC   CODE        LINE


6899  E4FC              ; ERROR MESSAGE TABLE
6900  E4FC              ; LEADING ERROR NUMBERS,
6901  E4FC              ; TEXT WITH 1ST & LAST CHARS
6902  E4FC              ; OR'ED WITH $80,
6903  E4FC              ; TOKENS FOR KEY WORDS ARE
6904  E4FC              ; LESS THAN $10 (AND'ED W/ $80)
6905  E4FC              ;
6906  E4FC              ERRTAB                 ; " OK"
6907  E4FC  00                 .BYTE 0,$A0,'O',$CB
6907  E4FD  A0
6907  E4FE  4F
6907  E4FF  CB
6908  E500                                    ;"READ ERROR"
6909  E500  20                 .BYTE $20,$21,$22,$23,$24,$27
6909  E501  21
6909  E502  22
6909  E503  23
6909  E504  24
6909  E505  27
6910  E506  D2                 .BYTE $D2,'EAD',$89
6910  E507  45 41 44
6910  E50A  89
6911  E50B                                    ;" FILE TOO LARGE"
6912  E50B  52                 .BYTE $52,$83,' TOO LARG',$C5
6912  E50C  83
6912  E50D  20 54
6912  E516  C5
6913  E517                                    ;" RECORD NOT PRESENT"
6914  E517  50                 .BYTE $50,$8B,6,' PRESEN',$D4
6914  E518  8B
6914  E519  06
6914  E51A  20 50
6914  E521  D4
6915  E522                                    ;"OVERFLOW IN RECORD"
6916  E522  51                 .BYTE $51,$CF,'VERFLOW '
6916  E523  CF
6916  E524  56 45
6917  E52C  49 4E              .BYTE 'IN',$8B
6917  E52E  8B
6918  E52F                                    ;" WRITE ERROR"
6919  E52F  25                 .BYTE $25,$28,$8A,$89
6919  E530  28
6919  E531  8A
6919  E532  89
6920  E533                                    ;" WRITE PROTECT ON"
6921  E533  26                 .BYTE $26,$8A,' PROTECT O',$CE
6921  E534  8A
6921  E535  20 50
6921  E53F  CE
6922  E540                                    ;" DISK ID MISMATCH"
6923  E540  29                 .BYTE $29,$88,' ID',$85
6923  E541  88
6923  E542  20 49 44
6923  E545  85
6924  E546                                    ;"SYNTAX ERROR"
```

```
LINE# LOC   CODE         LINE


6925  E546  30                  .BYTE $30,$31,$32,$33,$34
6925  E547  31
6925  E548  32
6925  E549  33
6925  E54A  34
6926  E54B  D3                  .BYTE $D3,'YNTAX',$89
6926  E54C  59 4E
6926  E551  89
6927  E552                                     ;" WRITE FILE OPEN"
6928  E552  60                  .BYTE $60,$8A,3,$84
6928  E553  8A
6928  E554  03
6928  E555  84
6929  E556                                     ;" FILE EXISTS"
6930  E556  63                  .BYTE $63,$83,' EXIST',$D3
6930  E557  83
6930  E558  20 45
6930  E55E  D3
6931  E55F                                     ;" FILE TYPE MISMATCH"
6932  E55F  64                  .BYTE $64,$83,' TYPE',$85
6932  E560  83
6932  E561  20 54
6932  E566  85
6933  E567                                     ;"NO BLOCK"
6934  E567  65                  .BYTE $65,$CE,'O BLOC',$CB
6934  E568  CE
6934  E569  4F 20
6934  E56F  CB
6935  E570                                     ;"ILLEGAL TRACK OR SECTOR"
6936  E570  66                  .BYTE $66,$67,$C9,'LLEGAL TRACK'
6936  E571  67
6936  E572  C9
6936  E573  4C 4C
6937  E57F  20 4F               .BYTE ' OR SECTO',$D2
6937  E588  D2
6938  E589                                     ;" FILE NOT OPEN"
6939  E589  61                  .BYTE $61,$83,6,$84
6939  E58A  83
6939  E58B  06
6939  E58C  84
6940  E58D                                     ;" FILE NOT FOUND"
6941  E58D  39                  .BYTE $39,$62,$83,6,$87
6941  E58E  62
6941  E58F  83
6941  E590  06
6941  E591  87
6942  E592                                     ;" FILES SCRATCHED"
6943  E592  01                  .BYTE 1,$83,'S SCRATCHE',$C4
6943  E593  83
6943  E594  53 20
6943  E59E  C4
6944  E59F                                     ;"NO CHANNEL"
6945  E59F  70                  .BYTE $70,$CE,'O CHANNE',$CC
6945  E5A0  CE
6945  E5A1  4F 20
```

```
LINE# LOC   CODE        LINE


6945  E5A9  CC
6946  E5AA                                    ;"DIR ERROR"
6947  E5AA  71                 .BYTE $71,$C4,'IR',$89
6947  E5AB  C4
6947  E5AC  49 52
6947  E5AE  89
6948  E5AF                                    ;" DISK FULL"
6949  E5AF  72                 .BYTE $72,$88,' FUL',$CC
6949  E5B0  88
6949  E5B1  20 46
6949  E5B5  CC
6950  E5B6                                    ;"CBM DOS V2.6 4030"
6951  E5B6  73                 .BYTE $73,$C3,'BM DOS V2.6 203',$B1
6951  E5B7  C3
6951  E5B8  42 4D
6951  E5C7  B1
6952  E5C8                                    ;"DRIVE NOT READY"
6953  E5C8  74                 .BYTE $74,$C4,'RIVE',6,' READ',$D9
6953  E5C9  C4
6953  E5CA  52 49
6953  E5CE  06
6953  E5CF  20 52
6953  E5D4  D9
6954  E5D5              ;
6955  E5D5              ; ERROR TOKEN KEY WORDS
6956  E5D5              ; WORDS USED MORE THAN ONCE
6957  E5D5                                    ;"ERROR"
6958  E5D5  09                 .BYTE 9,$C5,'RRO',$D2
6958  E5D6  C5
6958  E5D7  52 52 4F
6958  E5DA  D2
6959  E5DB                                    ;"WRITE"
6960  E5DB  0A                 .BYTE $A,$D7,'RIT',$C5
6960  E5DC  D7
6960  E5DD  52 49 54
6960  E5E0  C5
6961  E5E1                                    ;"FILE"
6962  E5E1  03                 .BYTE 3,$C6,'IL',$C5
6962  E5E2  C6
6962  E5E3  49 4C
6962  E5E5  C5
6963  E5E6                                    ;"OPEN"
6964  E5E6  04                 .BYTE 4,$CF,'PE',$CE
6964  E5E7  CF
6964  E5E8  50 45
6964  E5EA  CE
6965  E5EB                                    ;"MISMATCH"
6966  E5EB  05                 .BYTE 5,$CD,'ISMATC',$C8
6966  E5EC  CD
6966  E5ED  49 53
6966  E5F3  C8
6967  E5F4                                    ;"NOT"
6968  E5F4  06                 .BYTE 6,$CE,'O',$D4
6968  E5F5  CE
6968  E5F6  4F
```

```
LINE# LOC   CODE         LINE


6968  E5F7  D4
6969  E5F8                              ;"FOUND"
6970  E5F8  07                 .BYTE 7,$C6,'OUN',$C4
6970  E5F9  C6
6970  E5FA  4F 55 4E
6970  E5FD  C4
6971  E5FE                              ;"DISK"
6972  E5FE  08                 .BYTE 8,$C4,'IS',$CB
6972  E5FF  C4
6972  E600  49 53
6972  E602  CB
6973  E603                              ;"RECORD"
6974  E603  0B                 .BYTE $B,$D2,'ECOR',$C4
6974  E604  D2
6974  E605  45 43
6974  E609  C4
6975  E60A               ETEND  =*
```

```
LINE# LOC   CODE        LINE


6977  E60A              ; CONTROLLER ERROR ENTRY
6978  E60A              ; .A= ERROR #
6979  E60A              ; .X= JOB #
6980  E60A  48          ERROR  PHA
6981  E60B  86 F9              STX JOBNUM
6982  E60D  8A                 TXA
6983  E60E  0A                 ASL A
6984  E60F  AA                 TAX
6985  E610  B5 06              LDA HDRS,X     ; 4/12*****;RECALL TRACK,SECTOR
6986  E612  85 80              STA TRACK
6987  E614  B5 07              LDA HDRS+1,X   ; 4/12*********
6988  E616  85 81              STA SECTOR
6989  E618              ;
6990  E618  68                 PLA
6991  E619  29 0F              AND #$F        ; CONVERT CONTROLLER...
6992  E61B  F0 08              BEQ ERR1       ; ...ERRORS TO DOS ERRORS
6993  E61D  C9 0F              CMP #$F        ; CHECK NODRIVE ERROR
6994  E61F  D0 06              BNE ERR2
6995  E621              ;
6996  E621  A9 74              LDA #NODRIV
6997  E623  D0 08              BNE ERR3       ; BRA
6998  E625              ERR1
6999  E625  A9 06              LDA #6         ; CODE=16-->14
7000  E627  09 20       ERR2   ORA #$20
7001  E629  AA                 TAX
7002  E62A  CA                 DEX
7003  E62B  CA                 DEX
7004  E62C  8A                 TXA
7005  E62D              ERR3
7006  E62D  48                 PHA
7007  E62E  AD 2A 02           LDA CMDNUM
7008  E631  C9 00              CMP #VAL
7009  E633  D0 0F              BNE ERR4
7010  E635  A9 FF              LDA #$FF
7011  E637  8D 2A 02           STA CMDNUM
7012  E63A  68                 PLA
7013  E63B  20 C7 E6           JSR ERRMSG
7014  E63E  20 42 D0           JSR INITDR     ; INIT FOR VALIDATE
7015  E641  4C 48 E6           JMP CMDER3
7016  E644              ERR4
7017  E644  68                 PLA
7018  E645              CMDER2
7019  E645  20 C7 E6           JSR ERRMSG
7020  E648              CMDER3
7021  E648  20 BD C1           JSR CLRCB      ; CLEAR CMDBUF
7022  E64B  A9 00              LDA #0
7023  E64D  8D F9 02           STA WBAM       ; CLEAR AFTER ERROR
7024  E650  20 2C C1           JSR ERRON      ; SET ERROR LED
7025  E653  20 DA D4           JSR FREICH     ; FREE INTERNAL CHANNEL
7026  E656  A9 00              LDA #0         ; CLEAR POINTERS
7027  E658  85 A3              STA BUFTAB+CBPTR
7028  E65A  A2 45              LDX #TOPWRT
7029  E65C  9A                 TXS            ; PURGE STACK
7030  E65D  A5 84              LDA ORGSA
7031  E65F  29 0F              AND #$F
```

```
LINE# LOC   CODE         LINE


7032  E661  85 83               STA SA
7033  E663  C9 0F               CMP #$F
7034  E665  F0 31               BEQ ERR10
7035  E667  78                  SEI
7036  E668  A5 79               LDA LSNACT
7037  E66A  D0 1C               BNE LSNERR
7038  E66C  A5 7A               LDA TLKACT
7039  E66E  D0 10               BNE TLKERR
7040  E670               ;
7041  E670  A6 83               LDX SA
7042  E672  BD 2B 02            LDA LINTAB,X
7043  E675  C9 FF               CMP #$FF
7044  E677  F0 1F               BEQ ERR10
7045  E679  29 0F               AND #$F
7046  E67B  85 82               STA LINDX
7047  E67D  4C 8E E6            JMP TLERR
7048  E680               ;
7049  E680               ;
7050  E680               ; TALKER ERROR RECOVERY
7051  E680               ; IF COMMAND CHANNEL, RELEASE DAV
7052  E680               ; IF DATA CHANNEL, FORCE NOT READY
7053  E680               ; AND RELEASE CHANNEL
7054  E680  20 EB D0     TLKERR JSR FNDRCH
7055  E683  20 7C EA            JSR ITERR
7056  E686  D0 06               BNE TLERR      ; FINISH
7057  E688               ;
7058  E688               ; LISTENER ERROR RECOVERY
7059  E688               ; IF COMMAND CHANNEL, RELEASE RFD
7060  E688               ; IF DATA CHANNEL, FORCE NOT READY
7061  E688               ; AND RELEASE CHANNEL
7062  E688  20 07 D1     LSNERR JSR FNDWCH
7063  E68B  20 85 EA            JSR ILERR
7064  E68E               TLERR
7065  E68E  20 25 D1            JSR TYPFIL
7066  E691  C9 04               CMP #RELTYP
7067  E693  B0 03               BCS ERR10
7068  E695  20 27 D2            JSR FRECHN
7069  E698               ERR10
7070  E698  4C 3D EC            JMP IDLE
```

```
LINE# LOC   CODE        LINE


7072  E69B              ; CONVERT HEX TO BCD
7073  E69B  AA          HEXDEC TAX
7074  E69C  A9 00              LDA #0
7075  E69E  F8                 SED
7076  E69F  E0 00       HEX0   CPX #0
7077  E6A1  F0 07              BEQ HEX5
7078  E6A3  18                 CLC
7079  E6A4  69 01              ADC #1
7080  E6A6  CA                 DEX
7081  E6A7  4C 9F E6           JMP HEX0
7082  E6AA  D8          HEX5   CLD
7083  E6AB              ;
7084  E6AB              ; CONVERT BCD TO ASCII DEC
7085  E6AB              ; RETURN BCD IN .X
7086  E6AB              ; STORE ASCII IN (TEMP),Y
7087  E6AB  AA          BCDDEC TAX
7088  E6AC  4A                 LSR A
7089  E6AD  4A                 LSR A
7090  E6AE  4A                 LSR A
7091  E6AF  4A                 LSR A
7092  E6B0  20 B4 E6           JSR BCD2
7093  E6B3  8A                 TXA
7094  E6B4              BCD2
7095  E6B4  29 0F              AND #$F
7096  E6B6  09 30              ORA #$30
7097  E6B8  91 A5              STA (CB+2),Y
7098  E6BA  C8                 INY
7099  E6BB  60                 RTS
7100  E6BC              ;
7101  E6BC              ; TRANSFER ERROR MESSAGE TO
7102  E6BC              ; ERROR BUFFER
7103  E6BC              ;
7104  E6BC              OKERR
7105  E6BC  20 23 C1           JSR ERROFF
7106  E6BF  A9 00              LDA #0
7107  E6C1              ERRTS0
7108  E6C1  A0 00              LDY #0
7109  E6C3  84 80              STY TRACK
7110  E6C5  84 81              STY SECTOR
7111  E6C7              ;
7112  E6C7              ERRMSG
7113  E6C7  A0 00              LDY #0
7114  E6C9  A2 D5              LDX #<ERRBUF
7115  E6CB  86 A5              STX CB+2
7116  E6CD  A2 02              LDX #>ERRBUF
7117  E6CF  86 A6              STX CB+3
7118  E6D1  20 AB E6           JSR BCDDEC      ; CONVERT ERROR #
7119  E6D4  A9 2C              LDA #',
7120  E6D6  91 A5              STA (CB+2),Y
7121  E6D8  C8                 INY
7122  E6D9  AD D5 02           LDA ERRBUF
7123  E6DC  8D 43 02           STA CHNDAT+ERRCHN
7124  E6DF  8A                 TXA             ; ERROR # IN .X
7125  E6E0  20 06 E7           JSR ERMOVE      ; MOVE MESSAGE
7126  E6E3              ;
```

```
LINE# LOC   CODE        LINE


7127  E6E3  A9 2C       ERMSG2 LDA #',
7128  E6E5  91 A5              STA (CB+2),Y
7129  E6E7  C8                 INY
7130  E6E8  A5 80              LDA TRACK
7131  E6EA  20 9B E6           JSR HEXDEC     ; CONVERT TRACK #
7132  E6ED  A9 2C              LDA #',
7133  E6EF  91 A5              STA (CB+2),Y
7134  E6F1  C8                 INY
7135  E6F2  A5 81              LDA SECTOR     ; CONVERT SECTOR #
7136  E6F4  20 9B E6           JSR HEXDEC
7137  E6F7  88                 DEY
7138  E6F8  98                 TYA
7139  E6F9  18                 CLC
7140  E6FA  69 D5              ADC #<ERRBUF
7141  E6FC  8D 49 02           STA LSTCHR+ERRCHN ; SET LAST CHAR
7142  E6FF  E6 A5              INC CB+2
7143  E701  A9 88              LDA #RDYTLK
7144  E703  85 F7              STA CHNRDY+ERRCHN
7145  E705  60                 RTS
7146  E706                  ;
7147  E706                  ;**********************************;
7148  E706                  ;* ERMOVE - MOVE ERROR MESSAGE *;
7149  E706                  ;* FROM ERRTAB TO ERRBUF. *;
7150  E706                  ;* FULLY RECURSIVE FOR TOKEN *;
7151  E706                  ;* WORD PROSESSING. *;
7152  E706                  ;* INPUT: .A= BCD ERROR NUMBER *;
7153  E706                  ;**********************************;
7154  E706                  ;
7155  E706                  ERMOVE
7156  E706  AA                 TAX            ; SAVE .A
7157  E707  A5 86              LDA R0         ; SAVE R0,R0+1
7158  E709  48                 PHA
7159  E70A  A5 87              LDA R0+1
7160  E70C  48                 PHA
7161  E70D  A9 FC              LDA #<ERRTAB   ; SET POINTER TO TABLE
7162  E70F  85 86              STA R0
7163  E711  A9 E4              LDA #>ERRTAB
7164  E713  85 87              STA R0+1
7165  E715  8A                 TXA            ; RESTORE .A
7166  E716  A2 00              LDX #0         ; .X=0 FOR INDIRECT
7167  E718                  E10
7168  E718  C1 86              CMP (R0,X)     ; ?ERROR # = TABLE ENTRY?
7169  E71A  F0 21              BEQ E50        ; YES, SEND MESSAGE
7170  E71C                  ;
7171  E71C  48                 PHA            ; SAVE ERROR #
7172  E71D  20 75 E7           JSR EADV2      ; CHECK & ADVANCE PTR
7173  E720  90 05              BCC E30        ; MORE #'S TO CHECK
7174  E722                  E20
7175  E722  20 75 E7           JSR EADV2      ; ADVANCE PAST THIS MESSAGE

7176  E725  90 FB              BCC E20
7177  E727                  E30
7178  E727  A5 87              LDA R0+1       ; CHECK PTR
7179  E729  C9 E6              CMP #>ETEND
7180  E72B  90 08              BCC E40        ; <, CONTINUE
7181  E72D  D0 0A              BNE E45        ; >, QUIT
```

```
LINE# LOC   CODE        LINE


7182  E72F              ;
7183  E72F  A9 0A              LDA #<ETEND
7184  E731  C5 86              CMP R0
7185  E733  90 04              BCC E45        ; PAST END OF TABLE
7186  E735              E40
7187  E735  68                 PLA            ; RESTORE ERROR #
7188  E736  4C 18 E7           JMP E10        ; CHECK NEXT ENTRY
7189  E739              E45
7190  E739  68                 PLA            ; POP ERROR #
7191  E73A  4C 4D E7           JMP E90        ; GO QUIT
7192  E73D              ;
7193  E73D              E50                    ; THE NUMBER HAS BEEN LOCATED
7194  E73D  20 67 E7           JSR EADV1
7195  E740  90 FB              BCC E50        ; ADVANCE PAST OTHER #'S
7196  E742              E55
7197  E742  20 54 E7           JSR E60
7198  E745  20 67 E7           JSR EADV1
7199  E748  90 F8              BCC E55
7200  E74A              ;
7201  E74A  20 54 E7           JSR E60        ; CHECK FOR TOKEN OR LAST WORD
7202  E74D              E90
7203  E74D  68                 PLA            ; ALL FINISHED
7204  E74E  85 87              STA R0+1       ; RESTORE R0
7205  E750  68                 PLA
7206  E751  85 86              STA R0
7207  E753  60                 RTS
7208  E754              ;
7209  E754              E60
7210  E754  C9 20              CMP #$20       ; (MAX TOKEN #)+1
7211  E756  B0 0B              BCS E70        ; NOT A TOKEN
7212  E758  AA                 TAX
7213  E759  A9 20              LDA #$20       ; IMPLIED LEADING SPACE
7214  E75B  91 A5              STA (CB+2),Y
7215  E75D  C8                 INY
7216  E75E  8A                 TXA            ; RESTORE TOKEN #
7217  E75F  20 06 E7           JSR ERMOVE     ; ADD TOKEN WORD TO MESSAGE
                                               7218  E762  60
                                               RTS
7219  E763              E70
7220  E763  91 A5              STA (CB+2),Y   ; PUT CHAR IN MESSAGE
7221  E765  C8                 INY
7222  E766  60                 RTS
7223  E767              ;
7224  E767              ;ERROR ADVANCE & CHECK
7225  E767              ;
7226  E767              EADV1                  ; PRE-INCREMENT
7227  E767  E6 86              INC R0         ; ADVANCE PTR
7228  E769  D0 02              BNE EA10
7229  E76B  E6 87              INC R0+1
7230  E76D              EA10
7231  E76D  A1 86              LDA (R0,X)     ; GET CURRENT ENTRY
7232  E76F  0A                 ASL A          ; .C=1 IS END OR BEGINNING
7233  E770  A1 86              LDA (R0,X)
7234  E772  29 7F              AND #$7F       ; MASK OFF BIT7
7235  E774  60                 RTS
7236  E775              ;
```

```
LINE# LOC   CODE         LINE


7237  E775               EADV2                  ; POST-INCREMENT
7238  E775  20 6D E7            JSR EA10       ; CHECK TABLE ENTRY
7239  E778  E6 86               INC R0
7240  E77A  D0 02               BNE EA20
7241  E77C  E6 87               INC R0+1
7242  E77E               EA20
7243  E77E  60                  RTS
7244  E77F               ;
7245  E77F               ;.END
7245  E77F               ;
7246  E77F                      .LIB IEEINT
```

```
LINE# LOC   CODE        LINE


7248  E77F  AD 00 18    LE77F  LDA PB          ; GET BUS STATUS
7249  E782  09 40              ORA #DAV       ; SET DAV TRUE
7250  E784  8D 00 18           STA PB         ; ON BUS
7251  E787  AD 02 18           LDA DDRB1      ; SET ALL BUS LINES TO OUTPUT
7252  E78A  29 BF              AND #$FF-DAV   ; EXCEPT DAV
7253  E78C  8D 02 18           STA DDRB1      ; AND
7254  E78F  60                 RTS            ; EXIT
7255  E790              ; INITIALIZE THE IEEE BUS
7256  E790              ;
7257  E790  AD 01 18    BOOT   LDA PA1         ; IF DATA ON BUS
7258  E793  D0 EA              BNE LE77F      ; SET DAV TO INPUT
7259  E795  AD 02 18           LDA DDRB1      ; SET DAV VIA PIN
7260  E798  09 40              ORA #$40       ; TO INPUT
7261  E79A  8D 02 18           STA DDRB1
7262  E79D  AD 00 18           LDA PB         ; GET BUS STATUS LINES
7263  E7A0  09 40              ORA #DAV       ; FLOAT DAV LINE
7264  E7A2  8D 00 18           STA PB         ; ON BUS
7265  E7A5  AD 00 18           LDA PB         ; GET BUS LINE STATUS
7266  E7A8  29 02              AND #NRFD      ; IF NRFD
7267  E7AA  F0 D3              BEQ LE77F      ; TRUE, GO
7268  E7AC  AD 00 18           LDA PB         ; ELSE
7269  E7AF  29 BF              AND #$FF-DAV   ; SET DAV
7270  E7B1  8D 00 18           STA PB         ; FALSE
7271  E7B4  AD 00 18           LDA PB         ; GET BUS LINE STATUS
7272  E7B7  29 02              AND #NRFD      ; IF NRFD
7273  E7B9  D0 C4              BNE LE77F      ; TRUE, LOOP
7274  E7BB  AD 01 18    LE7BB  LDA PA1         ; GET BUS DATA
7275  E7BE  F0 FB              BEQ LE7BB      ; IF NONE, WAIT FOR SOME
7276  E7C0  EE 78 02           INC F2CNT
7277  E7C3  EE 74 02           INC CMDSIZ
7278  E7C6  A9 2A              LDA #$2A
7279  E7C8  8D 00 02           STA CMDBUF
7280  E7CB  20 7F E7           JSR LE77F      ; FLOAT DAV LINE
7281  E7CE  4C D6 E7           JMP BOOT4      ; AND BOOT CODE VIA UTLODR
7282  E7D1              ;
7283  E7D1              ; .END
7283  E7D1              ;
7284  E7D1                     .LIB UTLODR
```

```
LINE# LOC   CODE         LINE


7286  E7D1               ;********************************
7287  E7D1               ;*UTLODR-UTILITY LOADER USED TO
7288  E7D1               ; LOAD USER PROGRAMS OR SYSTEM
7289  E7D1               ; UTILITIES FROM DISK AND
7290  E7D1               ; EXECUTE THEM.
7291  E7D1               ;
7292  E7D1               ; THIS LOADER IS DESIGNED TO BE
7293  E7D1               ; INCORPORATED INTO THE 1540 DOS
7294  E7D1               ; SYSTEM (SERIAL BUS INTERFACE).
7295  E7D1               ;
7296  E7D1               ; HARDWARE REQIRED :
7297  E7D1               ; CONNECT DATA AND CLOCK LINE TO
7298  E7D1               ; GROUND. (2-4-5 ON CONNECTOR)
7299  E7D1               ;
7300  E7D1               ;********************************
7301  E7D1               ;*ON ENTRY-
7302  E7D1               ; ONLY REQUIREMENT IS THAT THE
7303  E7D1               ; FILENAME OF THE FILE TO BE
7304  E7D1               ; LOADED BE THE FIRST SPECIFIED
7305  E7D1               ; NAME IN CMDBUF (THE COMMAND
7306  E7D1               ; BUFFER).
7307  E7D1               ;
7308  E7D1               ; REGISTERS: IGNORED
7309  E7D1               ;
7310  E7D1               ;*ON EXIT-
7311  E7D1               ; IF THE FILE EXISTED ON DISK AND
7312  E7D1               ; COULD BE FOUND, AND NO CHECKSUM
7313  E7D1               ; ERRORS WERE ENCOUNTERED WHILE
7314  E7D1               ; LOADING IT, IT IS NOW LOADED
7315  E7D1               ; INTO MEMORY, READY TO EXECUTE.
7316  E7D1               ;
7317  E7D1               ; REGISTERS: ALL DESTROYED
7318  E7D1               ;
7319  E7D1               ; EXECUTION OF THE LOADED PROGRAM
7320  E7D1               ; IS STARTED AT THE FIRST BYTE
7321  E7D1               ; LOADED.
7322  E7D1               ;
7323  E7D1               ; CMDBUF CONTAINS THE PARAMETER
7324  E7D1               ; STRING FOR THE FRESHLY LOADED
7325  E7D1               ; UTILITY OR USER PROGRAM.
7326  E7D1               ;
7327  E7D1               ;********************************
```

```
LINE# LOC   CODE         LINE


7329  E7D1              ;********************************
7330  E7D1              ;
7331  E7D1              ;*FIRST WRITING-
7332  E7D1              ; 25-JAN-80
7333  E7D1              ; BY RON STEPHENS
7334  E7D1              ;
7335  E7D1              ;--------------------------------
7336  E7D1              ;
7337  E7D1              ;*REVISION LIST-
7338  E7D1              ; 28-FEB-80 S. PATTERSON - ADD PARSER INTERFACE
7339  E7D1              ;
7340  E7D1              ;********************************
```

```
LINE# LOC   CODE         LINE


7342  E7D1               ;********************************
7343  E7D1               ;
7344  E7D1               ;*ROUTINES EXTERNAL TO THIS
7345  E7D1               ; MODULE THAT ARE USED:
7346  E7D1               ;
7347  E7D1               ; LOOKUP-
7348  E7D1               ; FUNCTION-FINDS FIRST NAME
7349  E7D1               ; SPECIFIED IN CMDBUF IN THE
7350  E7D1               ; DISK DIRECTORY ON THE SPECIFIED
7351  E7D1               ; DRIVE.
7352  E7D1               ;
7353  E7D1               ; INPUTS FROM UTLODR-NONE
7354  E7D1               ; OUTPUTS TO UTLODR-NONE
7355  E7D1               ;
7356  E7D1               ; CHKIN-
7357  E7D1               ; FUNCTION-CHECKS IF NAME WAS
7358  E7D1               ; FOUND BY LOOKUP. ERROR IF NOT
7359  E7D1               ; FOUND. EXITS TO 8050 ERROR
7360  E7D1               ; ROUTINE.
7361  E7D1               ;
7362  E7D1               ; INPUTS FROM UTLODR-NONE
7363  E7D1               ; OUTPUTS TO UTLODR-NONE
7364  E7D1               ;
7365  E7D1               ; GIBYTE-
7366  E7D1               ; FUNCTION-FETCHES NEXT BYTE
7367  E7D1               ; FROM OPEN FILE. ALSO SETS
7368  E7D1               ; EOIFLG ZERO IF END OF FILE
7369  E7D1               ; CONDITION DETECTED.
7370  E7D1               ;
7371  E7D1               ; INPUTS FROM UTLODR-NONE
7372  E7D1               ; OUTPUTS TO UTLODR-VARIABLE 'DATA' CONTAINS
7373  E7D1               ; THE DATA BYTE.
7374  E7D1               ;
7375  E7D1               ; OPNTYP-
7376  E7D1               ; FUNCTION-OPENS FILE PREVIOUSLY
7377  E7D1               ; "LOOKED UP" BY LOOKUP ABOVE.
7378  E7D1               ; CALLS ERROR ROUTINE IF
7379  E7D1               ; TYPE DOESN'T MATCH THAT SPECIFIED
7380  E7D1               ;
7381  E7D1               ; INPUTS FROM UTLODR-.A=FILE TYPE
7382  E7D1               ; (5 IN OUR CASE)
7383  E7D1               ; OUTPUTS TO UTLODR-NONE
7384  E7D1               ;
7385  E7D1               ; CMDER2-
7386  E7D1               ; FUNCTION-PLACES ERROR MESSAGE
7387  E7D1               ; SPECIFIED INTO ERROR BUFFER.
7388  E7D1               ;
7389  E7D1               ; INPUTS FROM UTLODR-.A=ERROR NUMBER
7390  E7D1               ; OUTPUTS TO UTLODR-NONE
7391  E7D1               ;BOOT2
7392  E7D1               ; RTS ;EXIT
7393  E7D1               ; ;
7394  E7D1               ;BOOT ;POWER-ON DIAG SENSE LOADER
7395  E7D1               ; LDA PB ;GET PORT DATA
7396  E7D1               ; TAX ;SAVE FOR LATER
```

```
LINE# LOC   CODE        LINE


7397  E7D1              ; AND #CLKIN ;CHECK FOR CLK TO GND
7398  E7D1              ; BEQ BOOT2 ;NO...EXIT
7399  E7D1              ; TXA
7400  E7D1              ; AND #DATIN ;CHECK FOR DATA TO GND
7401  E7D1              ; BEQ BOOT2 ;NO...EXIT
7402  E7D1              ; CLI ;SO BACKGND WILL RUN!
7403  E7D1              ;
7404  E7D1              ;;BOOT CLIP MUST BE ON
7405  E7D1              ;BOOT3
7406  E7D1              ; LDA PB
7407  E7D1              ; AND #CLKIN+DATIN
7408  E7D1              ; BNE BOOT3 ;WAIT UNTILL REMOVED?
7409  E7D1              ;
7410  E7D1              ; INC F2CNT ;SET # FILES
7411  E7D1              ; INC CMDSIZ ;SET # OF CHARS
7412  E7D1              ; LDA #'*'
7413  E7D1              ; STA CMDBUF ;SET FILENAME FOR ANY MATCH
7414  E7D1              ; JMP BOOT4
7415  E7D1              ;
7416  E7D1              ;*ENTRY POINT
7417  E7D1              ;
7418  E7D1              UTLODR
7419  E7D1  A9 8D              LDA #$8D
7420  E7D3  20 68 C2           JSR PARSE
7421  E7D6              ;
7422  E7D6              BOOT4
7423  E7D6  20 A6 F2           JSR KILLP      ;KILL PROTECT
7424  E7D9  AD 78 02           LDA F2CNT
7425  E7DC  48                 PHA            ;SAVE FILE COUNT FOR UTILITY
7426  E7DD  A9 01              LDA #1
7427  E7DF  8D 78 02           STA F2CNT
7428  E7E2  A9 FF              LDA #$FF       ;INIT FIRSTBYTE FLAG
7429  E7E4  85 86              STA R0         ;R0 IS FLAG
7430  E7E6              ;
7431  E7E6  20 4F C4           JSR LOOKUP     ;LOCATE FILENAME ON DISK
7432  E7E9              ;
7433  E7E9  AD 80 02           LDA FILTRK     ;CHECK IF FOUND. ERR IF NOT
7434  E7EC  D0 05              BNE UTLD00
7435  E7EE  A9 39              LDA #NOCFIL
7436  E7F0  20 C8 C1           JSR CMDERR
7437  E7F3              UTLD00
7438  E7F3  68                 PLA
7439  E7F4  8D 78 02           STA F2CNT      ;RESTORE FILE COUNT
7440  E7F7              ;
7441  E7F7  AD 80 02           LDA FILTRK     ;INIT TRK, SCTR FOR OPEN
7442  E7FA  85 80              STA TRACK
7443  E7FC  AD 85 02           LDA FILSEC
7444  E7FF  85 81              STA SECTOR
7445  E801              ;
7446  E801  A9 03              LDA #USRTYP    ;OPEN SYSTEM TYPE FILE( 5 )
7447  E803  20 77 D4           JSR OPNTYP     ;OPEN
7448  E806              ;
7449  E806              ;*******************************
7450  E806              ;
7451  E806              ;*FILE RECORD FETCH LOOP
```

```
LINE# LOC   CODE        LINE


7452  E806  A9 00       UTLD10 LDA #$00        ;INIT CHECKSUM
7453  E808  85 87              STA R1         ;CALC. CHKSUM RESIDES IN R1
7454  E80A              ;
7455  E80A  20 67 E8           JSR GTABYT     ;FETCH LOAD ADDRESS LO
7456  E80D  85 88              STA R2
7457  E80F  20 79 E8           JSR ADDSUM     ;ADD INTO CHECKSUM
7458  E812              ;
7459  E812  20 67 E8           JSR GTABYT     ;FETCH LOAD ADDRESS HI
7460  E815  85 89              STA R3
7461  E817  20 79 E8           JSR ADDSUM
7462  E81A              ;
7463  E81A  A5 86              LDA R0         ;IS THIS THE FIRSTBYTE ADDRESS?
7464  E81C  F0 0A              BEQ UTLD20     ;BR IF NOT
7465  E81E  A5 88              LDA R2         ;SAV AWAY THIS ADDR. IN2 STACK
7466  E820  48                 PHA            ;LO FIRST
7467  E821  A5 89              LDA R3
7468  E823  48                 PHA            ;HI NEXT
7469  E824  A9 00              LDA #$00       ;CLEAR FLAG
7470  E826  85 86              STA R0         ;FIRSTBYTE FLAG
7471  E828              ;
7472  E828  20 67 E8    UTLD20 JSR GTABYT      ;FETCH DATA BYTE COUNT
7473  E82B  85 8A              STA R4         ;SAVE IN R4
7474  E82D  20 79 E8           JSR ADDSUM     ;ADD INTO CHECKSUM
7475  E830              ;
7476  E830              ;********************************
7477  E830              ;
7478  E830              ;*BYTE STORE LOOP
7479  E830  20 67 E8    UTLD30 JSR GTABYT      ;FETCH DATA BYT
7480  E833  A0 00              LDY #$00       ;INIT INDEX
7481  E835  91 88              STA (R2),Y     ;STORE BYTE AT DESIRED ADDRESS
7482  E837  20 79 E8           JSR ADDSUM     ;ADD INTO CHECKSUM
7483  E83A              ;
7484  E83A  A5 88              LDA R2         ;POINTER:=POINTER+1
7485  E83C  18                 CLC
7486  E83D  69 01              ADC #$01
7487  E83F  85 88              STA R2
7488  E841  90 02              BCC UTLD35
7489  E843  E6 89              INC R3         ;ADD IN CARRY
7490  E845              ;
7491  E845  C6 8A       UTLD35 DEC R4          ;UPDATE BYTE COUNTER
7492  E847  D0 E7              BNE UTLD30     ;IF NONZERO, CONTINUE
7493  E849              ;*END OF BYTE STORE LOOP
7494  E849              ;
7495  E849              ;********************************
7496  E849              ;
7497  E849  20 35 CA           JSR GIBYTE     ;GET A BYTE WITHOUT CHECK FOR EOI
7498  E84C  A5 85              LDA DATA
7499  E84E  C5 87              CMP R1         ;LAST BYTE FETCHED WAS CHKSUM
7500  E850  F0 08              BEQ UTLD50     ;IF SAME...EVERYTHING OK
7501  E852  20 3E DE           JSR GETHDR
7502  E855  A9 50              LDA #NOREC     ;SHOW RECORD OVERFLOW
7503  E857  20 45 E6           JSR CMDER2     ;AND LEAVE TO ERROR EXIT
7504  E85A              ;
7505  E85A  A5 F8       UTLD50 LDA EOIFLG      ;CHECK FOR END OF FILE
7506  E85C  D0 A8              BNE UTLD10     ;IF NONZERO, NOT DONE
```

```
LINE# LOC   CODE        LINE


7507  E85E              ;*END OF RECORD LOAD LOOP
7508  E85E              ;
7509  E85E              ;********************************
7510  E85E              ;
7511  E85E  68                 PLA            ;XFER CNTRL TO 1ST BYTE ADDR.
7512  E85F  85 89              STA R3
7513  E861  68                 PLA
7514  E862  85 88              STA R2
7515  E864  6C 88 00           JMP (R2)
```

```
LINE# LOC   CODE        LINE


7517  E867              ;********************************
7518  E867              ;
7519  E867              ;*LOCAL ROUTINES USED BY UTLODR
7520  E867              ;

7522  E867              ;********************************
7523  E867              ;
7524  E867              ;*GTABYT-FETCHES A BYTE FROM THE
7525  E867              ; FILE OPEN ON THE INTERNAL
7526  E867              ; CHANNEL. CHECKS IF THIS
7527  E867              ; WAS THE LAST BYTE IN THE
7528  E867              ; FILE. ERROR IF IT WAS.
7529  E867              ; SHOW A 'PTER' (PREMATURE
7530  E867              ; TERMINATION ERROR).
7531  E867              ;
7532  E867              ;********************************
7533  E867              ;
7534  E867              ;*ENTRY POINT
7535  E867              ;
7536  E867  20 35 CA    GTABYT JSR GIBYTE      ;FETCH A BYTE TO DATA LOC
7537  E86A  A5 F8              LDA EOIFLG     ;CHECK IF EOF EXISTS
7538  E86C  D0 08              BNE GTABYE     ;OK IF NONZERO
7539  E86E  20 3E DE           JSR GETHDR
7540  E871  A9 51              LDA #RECOVF    ;RECORD SIZE ERROR
7541  E873  20 45 E6           JSR CMDER2     ;CALL 8050 ERROR ROUTINE
7542  E876              ;
7543  E876  A5 85       GTABYE LDA DATA
7544  E878  60                 RTS


7546  E879              ;********************************
7547  E879              ;
7548  E879              ;*ADDSUM-ADDS UP CHECKSUM INTO
7549  E879              ; LOCATION R1. ALGORITHM:
7550  E879              ; NEWSUM:=OLDSUM+NEWBYTE+
7551  E879              ; CARRY
7552  E879              ;
7553  E879              ; INPUTS: EXPECTS NEWBYTE IN .A
7554  E879              ; OUTPUTS: R1=NEWSUM, .A DESTROYED
7555  E879              ;
7556  E879              ;********************************
7557  E879              ;
7558  E879              ;*ENTRY POINT
7559  E879              ;
7560  E879  18          ADDSUM CLC
7561  E87A  65 87              ADC R1         ;.A=.A+R1
7562  E87C  69 00              ADC #$00       ;.A=.A+CARRY
7563  E87E  85 87              STA R1         ;SAVE NEW CHECKSUM
7564  E880  60                 RTS
7565  E881              ;
7566  E881              ;.END
7566  E881              ;
7567  E881              ; ;.LIB PARATN
7568  E881              ; ;.LIB PARLISTN
7569  E881              ; ;.LIB PARTALK
```

```
LINE# LOC   CODE        LINE


7570  E881                     .LIB IEEESF
```

```
LINE# LOC   CODE        LINE


7572  E881              ;
7573  E881              ;THE IEEE BUS HANDSHAKE LINES ARE
7574  E881              ;CONTROLLED FROM A VIA PORT B AND CA1
7575  E881              ;USING THE FOLLOWING LAYOUT:
7576  E881              ;
7577  E881              ;PB0: ATN OUT
7578  E881              ;PB1: NRFD (I/O)
7579  E881              ;PB2: NDAC (I/O)
7580  E881              ;PB3: EOI (I/O)
7581  E881              ;PB4: TRANSMIT/RECEIVE CONTROL FOR BUFFER CHIPS
7582  E881              ;PB5: NOT USED FOR IEEE INTERFACE
7583  E881              ;PB6: DAV (I/O)
7584  E881              ;PB7 / CA1: ATN IN (CAUSES IRQS)
7585  E881              ;
7586  E881              ;
7587  E881              ;ATN IRQ PROCESS
7588  E881              ; IRQ ON ATN, LISTEN TO PET
7589  E881              ; CLEAR STACK


7591  E881  AD 01 18    ATNIRQ LDA IEEED       ; CLEAR IRQ FLAG
7592  E884  A9 01              LDA #1         ; SET ATN PENDING
7593  E886  85 7C              STA ATNPND     ; FLAG
7594  E888  60                 RTS            ; AND EXIT
7595  E889              ;
7596  E889              ;
7597  E889              ;
7598  E889  78          ATNSRV SEI
7599  E88A  A9 00              LDA #0         ; CLEAR PENDING FLAG
7600  E88C  85 7C              STA ATNPND
7601  E88E  A2 45              LDX #$45       ; INITIALIZE
7602  E890  9A                 TXS            ; STACK POINTER
7603  E891  A9 4C              LDA #DAV+EOI+NDAC
7604  E893  0D 00 18           ORA PB
7605  E896  29 ED              AND #$FF-NRFD-TR ; SET RECEIVE MODE
7606  E898  8D 00 18           STA PB
7607  E89B  A9 00              LDA #%00000000 ; DATA PORT IS
7608  E89D  8D 03 18           STA DDRA1      ; ALL INPUTS
7609  E8A0  A9 17              LDA #TR+NRFD+NDAC+ATNA ; SET PINS TO OUTPUTS
7610  E8A2  8D 02 18           STA DDRB1
7611  E8A5  A9 FF              LDA #$FF       ; DATA IS FALSE IN OUTPUT REGISTER
7612  E8A7  8D 01 18           STA IEEED
7613  E8AA  AD 00 18    ATN10  LDA PB          ; GET BUS LINES
7614  E8AD  29 FB              AND #$FF-NDAC  ; NDAC FALSE
7615  E8AF  09 03              ORA #ATNA+NRFD ; ATN ACK AND NRFD TRUE
7616  E8B1  8D 00 18           STA PB         ; SET BUS
7617  E8B4  2C 00 18    ATN20  BIT PB          ; GET BUS LINES
7618  E8B7  50 04              BVC ATN30      ; IF DAV, GO EVALUATE COMMAND
7619  E8B9  30 F9              BMI ATN20      ; IF ATN, WAIT FOR ATN RELEASE
7620  E8BB  10 7D              BPL ATN50      ; ELSE CHECK IF WERE LISTENING
                                               (BRANCH ALWAYS)
7621  E8BD              ;
7622  E8BD              ;ATN DROPPED, DAV TRUE, EVALUATE COMMAND
7623  E8BD  A9 FD       ATN30  LDA #$FF-NRFD   ; SET NRFD FALSE (=READY FOR
                                               DATA)
7624  E8BF  2D 00 18           AND PB         ; SET PORT DATA
```

```
LINE# LOC   CODE        LINE


7625  E8C2  8D 00 18           STA PB         ; SET BUS
7626  E8C5  29 08              AND #EOI       ; GET EOI BIT
7627  E8C7  85 F8              STA EOIFLG     ; SET EOI FLAG ACCORDINGLY
7628  E8C9  AD 01 18           LDA IEEED      ; GET BUS DATA
7629  E8CC  49 FF              EOR #$FF       ; INVERT IT
7630  E8CE  85 96              STA ICMD       ; AND STORE BUS COMMAND
7631  E8D0  A9 04              LDA #NDAC      ; GET BIT MASK
7632  E8D2  0D 00 18           ORA PB         ; SET NDAC TRUE
7633  E8D5  8D 00 18           STA PB         ; ON BUS
7634  E8D8  A0 00       DCDE   LDY #0          ; GET CLEAR STATE IN Y
7635  E8DA  A5 96              LDA ICMD       ; GET BUS COMMAND BYTE
7636  E8DC  29 60              AND #$60       ; GET COMMAND BITS
7637  E8DE  C9 40              CMP #$40       ; TALK?
7638  E8E0  F0 29              BEQ DCDE60     ; GO TALK
7639  E8E2  C9 20              CMP #$20       ; LISTEN?
7640  E8E4  F0 06              BEQ DCDE20     ; GO LISTEN
7641  E8E6  C9 60              CMP #$60       ; SECONADRY ADDRESS?
7642  E8E8  F0 2F              BEQ DCDE70     ; GO HANDLE SECONDARY ADDRESS
7643  E8EA  D0 46              BNE DCDE80     ; ELSE OTHER, DO THOSE
7644  E8EC              ;
7645  E8EC              ;LISTEN COMMAND RECEIVED
7646  E8EC  A5 96       DCDE20 LDA ICMD        ; GET BUS COMMAND
7647  E8EE  C5 77              CMP LSNADR     ; IS IT FOR US
7648  E8F0  F0 0B              BEQ DCDE40     ; YES, GO LISTEN
7649  E8F2  C9 3F              CMP #$3F       ; IF UNLISTEN
7650  E8F4  D0 02              BNE DCDE30
7651  E8F6  84 79              STY LSNACT     ; CLEAR LISTEN FLAG
7652  E8F8  84 7B       DCDE30 STY ADRSED      ; SET NOT ADDRESSED
7653  E8FA  4C 32 E9           JMP DCDE80     ; WAIT FOR NO DAV AND LOOP
7654  E8FD              ;
7655  E8FD              ; LISTEN TO IEEE488 BUS
7656  E8FD              ;
7657  E8FD  85 79       DCDE40 STA LSNACT      ; SAVE CURRENT LISTEN SA
7658  E8FF  84 7A              STY TLKACT     ; CLEAR CURRENT TALK ADDRESS
7659  E901  A9 20       DCDE50 LDA #$20        ; CURRENT STATE IS LISTEN
7660  E903  85 83              STA SA         ; SET IT
7661  E905  85 84              STA ORGSA
7662  E907  85 7B              STA ADRSED     ; SET LISTEN ADDRESS
7663  E909  D0 27              BNE DCDE80     ; WAIT FOR NO DAV AND LOOP (BRANCH
                                               ALWAYS)
7664  E90B              ;
7665  E90B              ; TALK TO IEEE448 BUS
7666  E90B              ;
7667  E90B  84 7A       DCDE60 STY TLKACT      ; CLEAR TALK ACTIVE FLAG
7668  E90D  A5 96              LDA ICMD       ; GET COMMAND BYTE
7669  E90F  C5 78              CMP TLKADR     ; IF NOT FOR US
7670  E911  D0 E5              BNE DCDE30     ; SET NOT ADDRESSED AND LOOP
7671  E913  85 7A              STA TLKACT     ; FOR US, FLAG WE'RE TALKING
7672  E915  84 79              STY LSNACT     ; CLEAR LISTENING FLAG
7673  E917  F0 E8              BEQ DCDE50     ; (BRANCH ALWAYS)
7674  E919              ;
7675  E919              ;HANDLE SECONDARY ADDRESS
7676  E919              ;
7677  E919  A5 7B       DCDE70 LDA ADRSED      ; IF NOT ADDRESSED
7678  E91B  F0 15              BEQ DCDE80     ; WAIT FOR NO DAV AND LOOP
7679  E91D  A5 96              LDA ICMD       ; ELSE GET COMMAND BYTE
```

```
LINE# LOC   CODE        LINE


7680  E91F  85 84              STA ORGSA      ; SET CURRENT SECONDARY ADD
RESS
7681  E921  48                 PHA            ; SAVE IT
7682  E922  29 0F              AND #$0F       ; STRIP THE COMMAND BITS
7683  E924  85 83              STA SA         ; AND SET AS TRUE SECONDARY
                                               ADDRESS
7684  E926  68                 PLA            ; RESTORE COMMAND BYTE
7685  E927  29 F0              AND #$F0       ; GET COMMAND BITS
7686  E929  C9 E0              CMP #$E0       ; IF CLOSE
7687  E92B  D0 05              BNE DCDE80
7688  E92D  58                 CLI            ; ALLOW INTERRUPTS
7689  E92E  20 C0 DA           JSR CLOSE      ; PERFORM CLOSE
7690  E931  78                 SEI            ; BLOCK INTERRUPTS, AND:
7691  E932              ; WAIT FOR NO DAV AND LOOP
7692  E932              DCDE80
7693  E932  2C 00 18    ATN40  BIT PB          ; WAIT
7694  E935  50 FB              BVC ATN40      ; FOR NO DAV
7695  E937  4C AA E8           JMP ATN10      ; AND LOOP
7696  E93A              ;
7697  E93A  A5 79       ATN50  LDA LSNACT      ; IF WE ARE NOT LISTENING
7698  E93C  F0 0E              BEQ ATN60      ; FLAG DATA ACCEPTED
7699  E93E  A9 FC              LDA #$FF-NRFD-ATNA ; ELSE SET BOTH NRFD AND ATN
7700  E940  2D 00 18           AND PB         ; FALSE
7701  E943  8D 00 18           STA PB         ; ON BUS
7702  E946  20 60 E9           JSR LISTEN     ; AND DO LISTEN
7703  E949  4C 3D EC           JMP IDLE       ; THEN IDLE BUS

7705  E94C  A9 FA       ATN60  LDA #$FF-NDAC-ATNA ; SET ATN AND NDAC
7706  E94E  2D 00 18           AND PB         ; FALSE (=DATA ACCEPTED)
7707  E951  09 04              ORA #NDAC      ; SET NDAC TRUE
7708  E953  8D 00 18           STA PB
7709  E956  A5 7A              LDA TLKACT     ; IF WE ARE NOT TALKING
7710  E958  F0 03              BEQ ATN70      ; GO IDLE BUS
7711  E95A  20 0A EA           JSR TALK       ; ELSE GO TALK
7712  E95D  4C 3D EC    ATN70  JMP IDLE        ; AND IDLE BUS
```

```
LINE# LOC   CODE        LINE


7714  E960              ;
7715  E960              ;
7716  E960              ;
7717  E960              LISTEN
7718  E960  78                 SEI            ; NO INTERRUPTS
7719  E961  A9 02              LDA #NRFD      ; SET NRFD
7720  E963  0D 00 18           ORA PB         ; FALSE (=READY FOR DATA)
7721  E966  8D 00 18           STA PB         ; ON BUS
7722  E969  2C 00 18    LSN10  BIT PB          ; GET BUS STATUS
7723  E96C  30 2A              BMI NOLATN     ; IF ATN, SERVICE THAT
7724  E96E  70 F9              BVS LSN10      ; IF DAV FALSE, WAIT FOR IT TO GO
                                               TRUE
7725  E970  20 07 D1           JSR FNDWCH     ; GET BUFFER NUMBER IN X
7726  E973  B0 05              BCS LSN15      ; IF NO BUFFER AVAILABLE

7728  E975  B5 F2              LDA CHNRDY,X   ; ELSE
7729  E977  6A                 ROR A          ; OPEN FOR LISTEN
7730  E978  B0 53              BCS LSN30
7731  E97A  A5 84       LSN15  LDA ORGSA       ; TEST IF CLOSE
7732  E97C  29 F0              AND #$F0       ; SECONDARY ADDRESS
7733  E97E  C9 F0              CMP #$F0       ; IS OPEN?
7734  E980  F0 4B              BEQ LSN30
7735  E982  A5 83       LSN20  LDA SA          ; GET SECONDARY ADDRESS
7736  E984  C9 01              CMP #$01       ; IF SAVING
7737  E986  F0 13              BEQ LSN25
7738  E988  2C 00 18    LSN21  BIT PB          ; ELSE GET BUS STATUS
7739  E98B  30 0B              BMI NOLATN     ; IF ATN, GO SERVICE THAT
7740  E98D  50 F9              BVC LSN21      ; IF DAV FALSE
7741  E98F  A9 FB              LDA #$FF-NDAC  ; SET NDAC FALSE
7742  E991  2D 00 18           AND PB         ; ON BUS
7743  E994  8D 00 18           STA PB         ; AND
7744  E997  60                 RTS            ; EXIT

7746  E998  4C 89 E8    NOLATN JMP ATNSRV      ; SERVICE ATN
7747  E99B              ;
7748  E99B  78          LSN25  SEI
7749  E99C  A9 FD              LDA #$FF-NRFD  ; NRFD FALSE (=READY FOR DATA)
7750  E99E  2D 00 18           AND PB         ; ADD TO BUS LIMES
7751  E9A1  8D 00 18           STA PB         ; AND SET BUS
7752  E9A4  A9 04              LDA #NDAC      ; NDAC FALE (=DATA ACCEPTED)
7753  E9A6  0D 00 18           ORA PB         ; SET BUS LINES
7754  E9A9  8D 00 18           STA PB         ; AND SET BUS
7755  E9AC  2C 00 18    LSN28  BIT PB          ; GET BUS STATE
7756  E9AF  30 E7              BMI NOLATN     ; IF ATN, GO SERVICE
7757  E9B1  50 F9              BVC LSN28      ; IF NO DAV, WAIT FOR IT
7758  E9B3  AD 00 18           LDA PB         ; IF DAV, GET BUS LINES
7759  E9B6  29 FB              AND #$FF-NDAC  ; SET NDAC
7760  E9B8  8D 00 18           STA PB         ; FALSE (=DATA ACCEPTED)
7761  E9BB  A9 02              LDA #NRFD      ; SET NRFD TRUE
7762  E9BD  0D 00 18           ORA PB         ; SET
7763  E9C0  8D 00 18           STA PB         ; PORT
7764  E9C3  2C 00 18    LSN29  BIT PB          ; IF ATN
7765  E9C6  30 D0              BMI NOLATN     ; GO SERVICE
7766  E9C8  50 F9              BVC LSN29      ; IF NO DAV, WAIT FOR IT
7767  E9CA  4C 9B E9           JMP LSN25      ; ELSE LOOP UNTIL ATN FALSE
7768  E9CD              ;
```

```
LINE# LOC   CODE        LINE


7769  E9CD  A9 FD       LSN30  LDA #$FF-NRFD   ; SET
7770  E9CF  2D 00 18           AND PB         ; NRFD TRUE
7771  E9D2  8D 00 18           STA PB         ; ON BUS
7772  E9D5  29 08              AND #EOI       ; GET EOI
7773  E9D7  85 F8              STA EOIFLG     ; STORE ITS STATUS
7774  E9D9  AD 01 18           LDA IEEED      ; GET BUS DATA
7775  E9DC  49 FF              EOR #$FF       ; INVERT IT
7776  E9DE  85 85              STA DATA       ; AND STORE BYTE
7777  E9E0  78                 SEI            ; NO INTERRUPTS
7778  E9E1  A9 04              LDA #NDAC      ; SET
7779  E9E3  0D 00 18           ORA PB         ; NDAC FALSE (=NOT ACCEPTED (YET))
7780  E9E6  8D 00 18           STA PB         ; ON BUS
7781  E9E9  2C 00 18    LSN40  BIT PB          ; IF ATN
7782  E9EC  30 AA              BMI NOLATN     ; GO SERVICE THAT
7783  E9EE  50 F9              BVC LSN40      ; IF NO DAV, WAIT FOR IT
7784  E9F0  A9 FB              LDA #$FF-NDAC  ; ELSE SET
7785  E9F2  2D 00 18           AND PB         ; NDAC FALSE (=DATA ACCEPTED)
7786  E9F5  8D 00 18           STA PB         ; ON BUS
7787  E9F8  58                 CLI            ; ALLOW INTERRUPTS
7788  E9F9  20 B7 CF           JSR PUT        ; BUT RECEIVED BYTE IN CHANNEL
7789  E9FC  4C 60 E9           JMP LISTEN     ; AND KEEP LISTENING


7791  E9FF  A9 48       TLK25  LDA #DAV+EOI    ; DAV FALSE, EOI FALSE
7792  EA01  0D 00 18           ORA PB         ; ADD CURRENT BUS LINES
7793  EA04  8D 00 18           STA PB         ; SET BUS
7794  EA07  4C 3D EC           JMP IDLE       ; AND IDLE BUS
```

```
LINE# LOC   CODE        LINE


7796  EA0A              ;
7797  EA0A              ;TALK TO IEEE488 BUS
7798  EA0A              ;
7799  EA0A              TALK
7800  EA0A  78                 SEI             ; NO INTERRUPTS
7801  EA0B  20 EB D0           JSR FNDRCH      ; FIND AN OPEN READ CHANNEL
7802  EA0E  B0 06              BCS NOTLK       ; TEST IF CHANNEL READY
7803  EA10  A6 82       TALK1  LDX LINDX        ; IF SO, GET INDEX
7804  EA12  B5 F2              LDA CHNRDY,X    ; GET BUFFER NUMBER
7805  EA14  30 01              BMI TLK10       ; IF READ CHANNEL, GO TALK
7806  EA16              NOTLK
7807  EA16  60          TLK05  RTS              ; ELSE EXIT
7808  EA17              ;
7809  EA17              ;CODE ADDED TO CORRECT VERIFY ERROR
7810  EA17  A9 59       TLK10  LDA #DAV+TR+EOI+ATNA ; SET OUTPUTS
7811  EA19  8D 02 18           STA DDRB1       ; SWITCH OUTPUTS
7812  EA1C  A9 FF              LDA #$FF        ; SET DATA
7813  EA1E  8D 03 18           STA DDRA1       ; TO OUTPUTS
7814  EA21  AD 00 18           LDA PB          ; GET BUS LINES
7815  EA24  09 10              ORA #TR         ; TRANSMIT MODE
7816  EA26  8D 00 18           STA PB          ; FOR BUFFERS
7817  EA29  A9 02              LDA #NRFD       ; GET MASK
7818  EA2B  2C 00 18    TLK20  BIT PB
7819  EA2E  30 49              BMI NOTATN      ; IF ATN, SERVICE IT
7820  EA30  F0 F9              BEQ TLK20       ; ELSE WAIT FOR ANY OTHER BUS LINE
                                                 TO GO TRUE
7821  EA32  BD 3E 02           LDA CHNDAT,X    ; THEN GET CHANNEL DATA
7822  EA35  49 FF              EOR #$FF        ; INVERT IT
7823  EA37  8D 01 18           STA IEEED       ; PUT ON BUS
7824  EA3A  B5 F2              LDA CHNRDY,X    ; GET CHANNEL STATE
7825  EA3C  09 B7              ORA #ATN+$20+TR+NRFD+NDAC+ATNA
7826  EA3E  2D 00 18           AND PB          ; ADD BUS FLAGS
7827  EA41  8D 00 18           STA PB          ; AND SET BUS
7828  EA44  AD 00 18    ISR04  LDA PB           ; GET BUS FLAGS
7829  EA47  30 30              BMI NOTATN      ; IF ATN, GO SERVICE IT
7830  EA49  29 06              AND #NRFD+NDAC ; ELSE GET NRFD AND NDAC
7831  EA4B  C9 06              CMP #NRFD+NDAC ; IF BOTH TRUE
7832  EA4D  F0 B0              BEQ TLK25
7833  EA4F  29 02              AND #NRFD       ; IF NRFD FALSE
7834  EA51  D0 F1              BNE ISR04       ; WAIT UNTIL NRFD TRUE & NDAC
                                                 FALSE
7835  EA53  58          TLK30  CLI              ; THEN:
7836  EA54  20 AA D3           JSR GET         ; GET A BYTE FROM CHANNEL
7837  EA57  78                 SEI             ; NO INTERRUPTS
7838  EA58  A9 04              LDA #NDAC       ; GET MASK FOR NDAC FALSE
7839  EA5A  2C 00 18    TLK35  BIT PB           ; GET BUS LINES
7840  EA5D  30 1A              BMI NOTATN      ; IF ATN, GO SERVICE IT
7841  EA5F  F0 F9              BEQ TLK35       ; IF NDAC TRUE, WAIT
7842  EA61  A9 FF       TLKRTN LDA #$FF         ; DATA ACCEPTED, CLEAR BUS DATA
7843  EA63  8D 01 18           STA IEEED       ; ON BUS
7844  EA66  A9 48              LDA #DAV+EOI    ; SET DAV & EOI TRUE
7845  EA68  0D 00 18           ORA PB          ; ADD CURRENT BUS STATE
7846  EA6B  8D 00 18           STA PB          ; SET BUS
7847  EA6E  A9 04              LDA #NDAC       ; GET MASK FOR NDAC
7848  EA70  2C 00 18    TLK40  BIT PB
7849  EA73  30 04              BMI NOTATN      ; IF ATN, GO SERVICE IT
7850  EA75  D0 F9              BNE TLK40       ; IF NDAC TRUE, WAIT FOR IT TO GO
                                                 FALSE
```

```
LINE# LOC   CODE        LINE


7851  EA77  F0 97              BEQ TALK1      ; AND KEEP ON TALKIN (BRANCH
                                                ALWAYS)
7852  EA79              ;
7853  EA79              NOTATN
7854  EA79  4C 89 E8           JMP ATNSRV     ; GO SERVICE ATN
7855  EA7C              ;
7856  EA7C              ITERR                  ; IEEE TALKER ERROR RECOVERY
7857  EA7C  AD 00 18           LDA PB         ; GET BUS LINES
7858  EA7F  09 40              ORA #DAV       ; SET DAV FALSE
7859  EA81  8D 00 18           STA PB         ; SET BUS
7860  EA84  60                 RTS            ; AND EXIT
7861  EA85              ;
7862  EA85              ILERR                  ; IEEE LISTENER ERROR RECOVERY
7863  EA85  A9 02              LDA #NRFD      ; GET MASK FOR NRFD TRUE
7864  EA87  0D 00 18           ORA PB         ; ADD TO BUS STATE
7865  EA8A  29 FE              AND #$FF-ATNA  ; SET ATN FALSE
7866  EA8C  8D 00 18           STA PB         ; SET BUS
7867  EA8F  60                 RTS            ; AND EXIT
7868  EA90              ;
7869  EA90              ;.END
7869  EA90              ;
7870  EA90                     .LIB DSKINTSF
```

```
LINE# LOC   CODE        LINE


7872  EA90              ;
7873  EA90              ;ERROR DISPLAY ROUTINE
7874  EA90              ;BLINKS THE (ERROR #)+1 IN ALL THREE LEDS
7875  EA90              ;
7876  EA90  A2 00       PEZRO  LDX #0           ;ERROR #1 FOR ZERO PAGE
7877  EA92  2C                 .BYTE $2C       ;SKIP NEXT INSTRUCTION
7878  EA93  A6 6F       PERR   LDX TEMP         ;GET ERROR #
7879  EA95  9A                 TXS             ;USE STACK AS STORAGE REG.
7880  EA96  BA          PE20   TSX              ;RESTORE ERROR #
7881  EA97  A9 08       PE30   LDA #LED0+LED1
7882  EA99  0D 00 1C           ORA LEDPRT
7883  EA9C  8D 00 1C           STA LEDPRT
7884  EA9F  8D 02 1C           STA DDRB2
7885  EAA2  98                 TYA             ;CLEAR INNER CTR
7886  EAA3  18          PD10   CLC
7887  EAA4  69 01       PD20   ADC #1           ;COUNT INNER CTR
7888  EAA6  D0 FC              BNE PD20
7889  EAA8  88                 DEY             ;DONE ?
7890  EAA9  D0 F8              BNE PD10        ;NO
7891  EAAB              ;
7892  EAAB  AD 00 1C           LDA LEDPRT
7893  EAAE  29 F7              AND #$FF-LED0-LED1
7894  EAB0  8D 00 1C           STA LEDPRT      ;TURN OFF ALL LEDS
7895  EAB3              PE40                    ;WAIT
7896  EAB3  98                 TYA             ;CLEAR INNER CTR
7897  EAB4  18          PD11   CLC
7898  EAB5  69 01       PD21   ADC #1           ;COUNT INNER CTR
7899  EAB7  D0 FC              BNE PD21
7900  EAB9  88                 DEY             ;DONE ?
7901  EABA  D0 F8              BNE PD11        ;NO
7902  EABC              ;
7903  EABC  CA                 DEX             ;BLINKED # ?
7904  EABD  10 D8              BPL PE30        ;NO - BLINK AGAIN
7905  EABF  E0 FC              CPX #$FC        ;WAITED BETWEEN COUNTS ?
7906  EAC1  D0 F0              BNE PE40        ;NO
7907  EAC3  F0 D1              BEQ PE20        ;ALWAYS - ALL AGAIN


7909  EAC5              DSKINT
7910  EAC5  78                 SEI             ;NO INTERRUPTS
7911  EAC6  D8                 CLD             ;CLEAR DECIMAL MODE
7912  EAC7  A2 FF              LDX #$FF        ;SET IEEE DATA PORT
7913  EAC9  8E 01 18           STX PA1         ;TO ZEROES ON BUS
7914  EACC  8E 03 18           STX DDRA1       ;AND TO INPUTS
7915  EACF  E8          DKIT10 INX              ;LET .X:=0
7916  EAD0              ;
7917  EAD0              ;
7918  EAD0              ;*********************************
7919  EAD0              ;
7920  EAD0              ; POWER UP DIAGNOSTIC
7921  EAD0              ;
7922  EAD0              ;*********************************
7923  EAD0              ;
7924  EAD0              ;
```

```
LINE# LOC   CODE        LINE


7925  EAD0  A9 CE              LDA #%11001110
7926  EAD2  8D 00 18           STA PB
7927  EAD5  A9 31              LDA #%00110001
7928  EAD7  8D 02 18           STA DDRB1
7929  EADA  A0 00              LDY #$00
7930  EADC  A2 00              LDX #$00       ;CLEAR ZERO PAGE
7931  EADE  8A          PU10   TXA             ;FILL Z-PAGE ACCEND PATTERN
7932  EADF  95 00              STA $00,X
7933  EAE1  E8                 INX
7934  EAE2  D0 FA              BNE PU10
7935  EAE4  8A          PU20   TXA             ;CHECK PATTERN BY INC...
7936  EAE5  D5 00              CMP $0,X       ;...BACK TO ORIG #
7937  EAE7  D0 A7              BNE PEZRO      ;BAD BITS, BLINK LEDS
7938  EAE9              PU30
7939  EAE9  F6 00              INC $0,X       ;BUMP CONTENTS
7940  EAEB  C8                 INY
7941  EAEC  D0 FB              BNE PU30       ;NOT DONE
7942  EAEE              ;
7943  EAEE  D5 00              CMP $0,X       ;CHECK FOR GOOD COUNT
7944  EAF0  D0 9E              BNE PEZRO      ;SOMETHING'S WRONG
7945  EAF2              ;
7946  EAF2  94 00              STY $0,X       ;LEAVE Z-PAGE ZEROED
7947  EAF4  B5 00              LDA $0,X       ;CHECK IT
7948  EAF6  D0 98              BNE PEZRO      ;WRONG
7949  EAF8              ;
7950  EAF8  E8                 INX            ;NEXT!
7951  EAF9  D0 E9              BNE PU20       ;NOT ALL DONE
7952  EAFB              ;
7953  EAFB              ;TEST TWO 64K-BIT ROMS
7954  EAFB              ;
7955  EAFB              ;ENTER X=START PAGE
7956  EAFB              ;EXIT IF OK
7957  EAFB              ;
7958  EAFB  E6 6F       RM10   INC TEMP        ;NEXT ERROR #
7959  EAFD  86 76              STX IP+1       ;SAVE PAGE, START X=0
7960  EAFF  A9 00              LDA #0
7961  EB01  85 75              STA IP         ;ZERO LO INDIRECT
7962  EB03  A8                 TAY
7963  EB04  A2 20              LDX #32        ;32 PAGES IN 8K ROM
7964  EB06  18                 CLC
7965  EB07  C6 76       RT10   DEC IP+1        ;DO IT BACKWARDS
7966  EB09  71 75       RT20   ADC (IP),Y      ;TOTAL CHECKSUM IN A
7967  EB0B  C8                 INY
7968  EB0C  D0 FB              BNE RT20
7969  EB0E  CA                 DEX
7970  EB0F  D0 F6              BNE RT10
7971  EB11  69 00              ADC #0         ;ADD IN LAST CARRY
7972  EB13              ; STA $0, ;ADJ VALUE AT $FFE5, GET VALUE $C001, CHG TO
                                               ADC #0,
7973  EB13              ; ADD +28 TO $FFE5, DONE !
7974  EB13  AA                 TAX            ;SAVE LOWER PAGE IN X
7975  EB14  C5 76              CMP IP+1       ;CORRECT ?
7976  EB16              ;--------CHECKSUM FOR DEBUG--------
7977  EB16  D0 39              BNE PERR2      ;NO - SHOW ERROR NUMBER
7978  EB18              ;----------------------------------
7979  EB18  E0 C0              CPX #$C0       ; ALL ROMS DONE?
```

```
LINE# LOC   CODE        LINE


7980  EB1A  D0 DF              BNE RM10       ; LOOP IF NOT

7982  EB1C              ;TEST ALL COMMON RAM
7983  EB1C              ;
7984  EB1C  A9 01       CR20   LDA #$01        ;START OF 1ST BLOCK
7985  EB1E  85 76       CR30   STA IP+1        ;SAVE PAGE #
7986  EB20  E6 6F              INC TEMP       ;BUMP ERROR #
7987  EB22              ;ENTER X=# OF PAGES IN BLOCK
7988  EB22              ; IP PTR TO FIRST PAGE IN BLOCK
7989  EB22              ;EXIT IF OK
7990  EB22              ;
7991  EB22  A2 07       RAMTST LDX #7          ;SAVE PAGE COUNT
7992  EB24  98          RA10   TYA             ;FILL WITH ADR SENSITIVE PATTERN
7993  EB25  18                 CLC
7994  EB26  65 76              ADC IP+1
7995  EB28  91 75              STA (IP),Y
7996  EB2A  C8                 INY
7997  EB2B  D0 F7              BNE RA10
7998  EB2D  E6 76              INC IP+1
7999  EB2F  CA                 DEX
8000  EB30  D0 F2              BNE RA10
8001  EB32  A2 07              LDX #7         ;RESTORE PAGE COUNT
8002  EB34  C6 76       RA30   DEC IP+1        ;CHECK PATTERN BACKWARDS
8003  EB36  88          RA40   DEY
8004  EB37  98                 TYA            ;GEN PATTERN AGAIN
8005  EB38  18                 CLC
8006  EB39  65 76              ADC IP+1
8007  EB3B  D1 75              CMP (IP),Y     ;OK ?
8008  EB3D  D0 12              BNE PERR2      ;NO - SHOW ERROR #
8009  EB3F  49 FF              EOR #$FF       ;YES - TEST INVERSE PATTERN
                                               8010  EB41  91 75
                                               STA (IP),Y
8011  EB43  51 75              EOR (IP),Y     ;OK ?
8012  EB45  91 75              STA (IP),Y     ;LEAVE MEMORY ZERO
8013  EB47  D0 08              BNE PERR2      ;NO - SHOW ERROR #
8014  EB49  98                 TYA
8015  EB4A  D0 EA              BNE RA40
8016  EB4C  CA                 DEX
8017  EB4D  D0 E5              BNE RA30
8018  EB4F              ;
8019  EB4F  F0 03              BEQ DIAGOK     ;BRANCH ALWAYS
8020  EB51              ;
8021  EB51  4C 93 EA    PERR2  JMP PERR

8023  EB54              ;
8024  EB54              DIAGOK
8025  EB54  A2 45              LDX #TOPWRT    ;INITIALIZE
8026  EB56  9A                 TXS            ;STACK POINTER
8027  EB57  AD 00 1C           LDA LEDPRT     ;SWITCH
8028  EB5A  29 F7              AND #LED0+239  ;DRIVE LED
8029  EB5C  8D 00 1C           STA LEDPRT     ;OFF
8030  EB5F  A9 01              LDA #%00000001 ;NEG EDGE OF ATN
```

```
LINE# LOC   CODE        LINE


8031  EB61  8D 0C 18           STA PCR1
8032  EB64  A9 82              LDA #%10000010 ;EABLE CA1 IRQS
8033  EB66  8D 0D 18           STA IFR1       ;CLEAR IRQ FLAGS
8034  EB69  8D 0E 18           STA IER1       ;IRQ ON CA1 NEG. EDGE
8035  EB6C  AD 00 18           LDA PB         ;GET IEEE BUS LINES
8036  EB6F  09 10              ORA #%00010000 ;SET TRANSMIT MODE FOR BUFFERS
8037  EB71  8D 00 18           STA PB         ;IN VIA PORT
8038  EB74  AD 0C 18           LDA PCR1       ;GET PCR
8039  EB77              ;CA1: POSITIVE TRANSITION,
8040  EB77              ;CA2=OUT, LOW,
8041  EB77              ;CB1: POSITVE TRANSITION,
8042  EB77              ;CB2=OUT, HIGH
8043  EB77  09 0C              ORA #%00001100 ;SET
8044  EB79  29 FD              AND #%11111101 ;FINAL VALUES
8045  EB7B  8D 0C 18           STA PCR1       ;IN CONTROL REGISTER
8046  EB7E  AD 02 18           LDA DDRB1      ;IEEE BUSLINE CONTROL,
8047  EB81  29 FE              AND #%11111110 ;SET ATN TO INPUT, REST UNCHANGED
8048  EB83  8D 02 18           STA DDRB1
8049  EB86  AD 00 18           LDA PB         ;COMPUTE PRIMARY ADDR
8050  EB89  29 03              AND #%00000011 ;PB0 AND PB1 ARE INPUTS
8051  EB8B  09 48              ORA #%01001000 ;TALK ADDRESS: $40 + 8 + INPUT
                                               SETTING
8052  EB8D  85 78              STA TLKADR     ;SET OUR TALK ADDRESS
8053  EB8F  49 60              EOR #$60       ;LISTEN ADDRESS
8054  EB91  85 77              STA LSNADR     ;SET OUR LISTEN ADDRESS
8055  EB93  AD 0C 18           LDA PCR1       ;RAISE
8056  EB96  09 02              ORA #%00000010 ;CA2
8057  EB98  8D 0C 18           STA PCR1       ;ON IEEE VIA (NO READING OF DIODE
                                               SETTING)
8058  EB9B  AD 00 18           LDA PB         ;GET PORT B DATA
8059  EB9E  29 EE              AND #%11101110 ;SET RECEIVE MODE
8060  EBA0  8D 00 18           STA PB         ;FOR BUFFERS
8061  EBA3  AD 02 18           LDA DDRB1      ;SET ATN PIN
8062  EBA6  09 01              ORA #%00000001 ;TO OUTPUT
8063  EBA8  8D 02 18           STA DDRB1
8064  EBAB                  ;
8065  EBAB                  ;INITIALIZE BUFFER PNTR TABLE
8066  EBAB                  ;
8067  EBAB  A2 00       INTTAB LDX #0
8068  EBAD  A0 00              LDY #0
8069  EBAF  A9 00       INTT1  LDA #0
8070  EBB1  95 99              STA BUFTAB,X
8071  EBB3  E8                 INX
8072  EBB4  B9 2F FF           LDA BUFIND,Y
8073  EBB7  95 99              STA BUFTAB,X
8074  EBB9  E8                 INX
8075  EBBA  C8                 INY
8076  EBBB  C0 05              CPY #BFCNT
8077  EBBD  D0 F0              BNE INTT1
8078  EBBF                  ;
8079  EBBF  A9 00              LDA #<CMDBUF   ;SET PNTR TO CMDBUF
8080  EBC1  95 99              STA BUFTAB,X
8081  EBC3  E8                 INX
8082  EBC4  A9 02              LDA #>CMDBUF
8083  EBC6  95 99              STA BUFTAB,X
8084  EBC8  E8                 INX
8085  EBC9  A9 D5              LDA #<ERRBUF   ;SET PNTR TO ERRBUF
```

```
LINE# LOC   CODE        LINE


8086  EBCB  95 99              STA BUFTAB,X
8087  EBCD  E8                 INX
8088  EBCE  A9 02              LDA #>ERRBUF
8089  EBD0  95 99              STA BUFTAB,X
8090  EBD2              ;
8091  EBD2  A9 FF              LDA #$FF
8092  EBD4  A2 12              LDX #MAXSA
8093  EBD6  9D 2B 02    DSKIN1 STA LINTAB,X
8094  EBD9  CA                 DEX
8095  EBDA  10 FA              BPL DSKIN1
8096  EBDC              ;
8097  EBDC  A2 05              LDX #MXCHNS-1
8098  EBDE              DSKIN2
8099  EBDE  95 A7              STA BUF0,X     ;SET BUFFERS AS UNUSED
8100  EBE0  95 AE              STA BUF1,X
8101  EBE2  95 CD              STA SS,X
8102  EBE4  CA                 DEX
8103  EBE5  10 F7              BPL DSKIN2
8104  EBE7              ;
8105  EBE7  A9 05              LDA #BFCNT     ;SET BUFFER POINTERS
8106  EBE9  85 AB              STA BUF0+CMDCHN
8107  EBEB  A9 06              LDA #BFCNT+1
8108  EBED  85 AC              STA BUF0+ERRCHN
8109  EBEF  A9 FF              LDA #$FF
8110  EBF1  85 AD              STA BUF0+BLINDX
8111  EBF3  85 B4              STA BUF1+BLINDX

8113  EBF5  A9 05              LDA #ERRCHN
8114  EBF7  8D 3B 02           STA LINTAB+ERRSA
8115  EBFA  A9 84              LDA #CMDCHN+$80
8116  EBFC  8D 3A 02           STA LINTAB+CMDSA
8117  EBFF  A9 0F              LDA #LXINT     ;LINDX 0 TO 5 FREE
8118  EC01  8D 56 02           STA LINUSE

8120  EC04  A9 01              LDA #RDYLST
8121  EC06  85 F6              STA CHNRDY+CMDCHN
8122  EC08  A9 88              LDA #RDYTLK
8123  EC0A  85 F7              STA CHNRDY+ERRCHN
8124  EC0C  A9 E0              LDA #$E0
8125  EC0E  8D 4F 02           STA BUFUSE
8126  EC11  A9 FF              LDA #$FF
8127  EC13  8D 50 02           STA BUFUSE+1
8128  EC16  A9 01              LDA #1
8129  EC18  85 1C              STA WPSW
8130  EC1A  85 1D              STA WPSW+1
8131  EC1C  20 63 CB           JSR USRINT     ;INIT USER JMP
8132  EC1F  20 FA CE           JSR LRUINT
8133  EC22              ;
8134  EC22              ;*********************************
8135  EC22              ;
8136  EC22              ; CONTROLLER INITIALIZATION
8137  EC22              ;
8138  EC22              ;*********************************
8139  EC22              ;
8140  EC22  20 A7 F2           JSR CNTINT
```

```
LINE# LOC   CODE        LINE


8142  EC25              ; SET INDIRECT VECTORS
8143  EC25  A9 54              LDA #<DIAGOK
8144  EC27  85 65              STA VNMI
8145  EC29  A9 EB              LDA #>DIAGOK
8146  EC2B  85 66              STA VNMI+1
8147  EC2D              ;
8148  EC2D  A9 0A              LDA #10        ;SET UP SECTOR OFFSET
8149  EC2F  85 69              STA SECINC
8150  EC31  A9 05              LDA #5
8151  EC33  85 6A              STA REVCNT     ;SET UP RECOVERY COUNT

8153  EC35              ;*
8154  EC35              ;*******************************
8155  EC35              ;*
8156  EC35              ;* SETERR
8157  EC35              ;* SET UP POWER ON ERROR MSG
8158  EC35              ;*
8159  EC35              ;* CBM DOS V2.0 (C)1979
8160  EC35              ;*
8161  EC35              ;*******************************
8162  EC35              ;*
8163  EC35              ;*
8164  EC35              ;
8165  EC35  A9 73       SETERR LDA #$73
8166  EC37  20 C1 E6           JSR ERRTS0
8167  EC3A              ;
8168  EC3A              ;
8169  EC3A              ;MUST BE CONTIGUOUS TO .FILE IDLE
8170  EC3A              ;
8171  EC3A              ;********************************
8172  EC3A              ; INIT THE IEEE BUS
8173  EC3A              ;
8174  EC3A              ;********************************
8175  EC3A              ;
8176  EC3A              ;---------------------------------------
8177  EC3A  20 90 E7           JSR BOOT
8178  EC3D              ;
8179  EC3D              ; .END
8179  EC3D              ;
8180  EC3D                     .LIB IDLESF
```

```
LINE# LOC   CODE        LINE


8183  EC3D              ; IDLE LOOP, WAITING FOR SOMETHING TO DO
8184  EC3D              ;*****************************************
8185  EC3D              ;
8186  EC3D              ; IEEE 488 CODE: DIFFERENT!
8187  EC3D              ;
8188  EC3D              ;*****************************************
8189  EC3D              IDLE
8190  EC3D  58                 CLI
8191  EC3E              ;
8192  EC3E  AD 55 02           LDA CMDWAT     ;TEST FOR PENDING COMMAND
8193  EC41  F0 0A              BEQ IDL1       ;NO COMMAND WAITING
8194  EC43  A9 00              LDA #0
8195  EC45  8D 55 02           STA CMDWAT
8196  EC48  85 67              STA NMIFLG     ;CLEAR DEBOUNCE
8197  EC4A  20 46 C1           JSR PARSXQ     ; PARSE AND XEQ COMMAND
8198  EC4D  58          IDL1   CLI             ;TEST FOR DRIVE RUNNING OR OPEN
                                               FILE
8199  EC4E  A5 7C              LDA ATNPND     ;IF NO ATN PENDING
8200  EC50  F0 03              BEQ IDL01      ;WAIT FOR COMMAND
8201  EC52  4C 89 E8           JMP ATNSRV     ;SERVICE ATN IRQ
8202  EC55              ;
8203  EC55              ; ENTER WAIT LOOP; WAITS FOR A COMMAND
8204  EC55              IDL01
8205  EC55  58                 CLI
8206  EC56  A9 0E              LDA #14
8207  EC58  85 72              STA TEMP+3
8208  EC5A  A9 00              LDA #0         ;IF FILE OPEN, TURN ON ACT LED
8209  EC5C  85 6F              STA TEMP
8210  EC5E  85 70              STA TEMP+1
8211  EC60  A6 72       IDL2   LDX TEMP+3      ;LOOK THRU LINTAB
8212  EC62  BD 2B 02           LDA LINTAB,X   ;FOR ACTIVE FILE
8213  EC65  C9 FF              CMP #$FF
8214  EC67  F0 10              BEQ IDL3
8215  EC69  29 3F              AND #$3F
8216  EC6B  85 82              STA LINDX
8217  EC6D  20 93 DF           JSR GETACT
8218  EC70  AA                 TAX
8219  EC71  BD 5B 02           LDA LSTJOB,X   ;DETERMINE WHICH DRIVE IT IS ON
8220  EC74  29 01              AND #1
8221  EC76  AA                 TAX
8222  EC77  F6 6F              INC TEMP,X
8223  EC79  C6 72       IDL3   DEC TEMP+3      ;SET FLAG INDICATING DRV
8224  EC7B  10 E3              BPL IDL2       ;HAS FILE OPEN
8225  EC7D  A0 04              LDY #BFCNT-1   ;LOOK THRU JOB QUE FOR
8226  EC7F  B9 00 00    IDL4   LDA JOBS,Y      ; FOR JOBS STILL RUNNING
8227  EC82  10 05              BPL IDL5
8228  EC84  29 01              AND #1
8229  EC86  AA                 TAX
8230  EC87  F6 6F              INC TEMP,X     ;SET FLAG INDICATING DRIVE
8231  EC89  88          IDL5   DEY             ;IS ACTIVE
8232  EC8A  10 F3              BPL IDL4
8233  EC8C  78                 SEI
8234  EC8D              ; DO NOT ALLOW IRQ WHEN READING LEDPRT
                                               *********************
```

```
LINE# LOC   CODE        LINE


8235  EC8D  AD 00 1C           LDA LEDPRT
8236  EC90  29 F7              AND #$FF-LED0
8237  EC92  48                 PHA
8238  EC93  A5 7F              LDA DRVNUM
8239  EC95  85 86              STA R0
8240  EC97  A9 00              LDA #0
8241  EC99  85 7F              STA DRVNUM
8242  EC9B  A5 6F              LDA TEMP
8243  EC9D  F0 0B              BEQ IDL7
8244  EC9F  A5 1C              LDA WPSW
8245  ECA1  F0 03              BEQ IDL6
8246  ECA3  20 13 D3           JSR CLDCHN
8247  ECA6              IDL6
8248  ECA6  68                 PLA            ;TURN ON LED IF DRIVE FLAG
8249  ECA7  09 08              ORA #LED0
8250  ECA9  48                 PHA
8251  ECAA              IDL7
8252  ECAA  E6 7F              INC DRVNUM
8253  ECAC  A5 70              LDA TEMP+1
8254  ECAE  F0 0B              BEQ IDL9
8255  ECB0  A5 1D              LDA WPSW+1
8256  ECB2  F0 03              BEQ IDL8
8257  ECB4  20 13 D3           JSR CLDCHN
8258  ECB7              IDL8
8259  ECB7  68                 PLA
8260  ECB8  09 00              ORA #LED1
8261  ECBA  48                 PHA
8262  ECBB              IDL9
8263  ECBB  A5 86              LDA R0
8264  ECBD  85 7F              STA DRVNUM
8265  ECBF  68                 PLA
8266  ECC0  AE 6C 02           LDX ERWORD
8267  ECC3  F0 21              BEQ IDL12      ;NO ERROR FLASHING
8268  ECC5              ;
8269  ECC5  AD 00 1C           LDA LEDPRT     ;USE CURRENT LEDS
8270  ECC8  E0 80              CPX #$80
8271  ECCA  D0 03              BNE IDL10      ;NOT IST TIME
8272  ECCC              ;
8273  ECCC              ;PHA
8274  ECCC              ;BEEP HERE**************
8275  ECCC              ;PLA
8276  ECCC  4C D9 EC           JMP IDL11
8277  ECCF              ;
8278  ECCF              IDL10
8279  ECCF  AE 05 18           LDX TIMER1
8280  ECD2  30 12              BMI IDL12      ;STILL TIMING
8281  ECD4              ;
8282  ECD4  A2 A0              LDX #$A0       ;COUNT 8 MS
8283  ECD6  8E 05 18           STX TIMER1
8284  ECD9              IDL11
8285  ECD9  CE 6C 02           DEC ERWORD     ;COUNT UNITS OF 8 MSEC
8286  ECDC  D0 08              BNE IDL12      ;KEEP COUNTING
8287  ECDE              ;
8288  ECDE  4D 6D 02           EOR ERLED      ;TOGGLE LED
8289  ECE1  A2 10              LDX #16        ;COUNT 16 UNITS
```

```
LINE# LOC   CODE        LINE


8290  ECE3  8E 6C 02           STX ERWORD
8291  ECE6              IDL12
8292  ECE6  8D 00 1C           STA LEDPRT      ;SET LEDS
8293  ECE9  4C 4D EC           JMP IDL1        ;BACK TO TOP OF LOP
8294  ECEC              ;
8295  ECEC              ; .END
8295  ECEC              ;
8296  ECEC                     .LIB LSTDIR
```

```
LINE# LOC   CODE        LINE


8298  ECEC              ; START THE DIRECTORY LOADING FUNCTION
8299  ECEC              ; GET THE BUFFER AND GET IT STARTED
8300  ECEC              ;
8301  ECEC  A9 00       STDIR  LDA #0
8302  ECEE  85 83              STA SA
8303  ECF0  A9 01              LDA #1         ; ALLOCATE CHANL AND 1 BUFEFER
8304  ECF2  20 E2 D1           JSR GETRCH

8306  ECF5  A9 00              LDA #0
8307  ECF7  20 C8 D4           JSR SETPNT

8309  ECFA  A6 82              LDX LINDX
8310  ECFC  A9 00              LDA #0
8311  ECFE  9D 44 02           STA LSTCHR,X
8312  ED01  20 93 DF           JSR GETACT
8313  ED04  AA                 TAX
8314  ED05  A5 7F              LDA DRVNUM
8315  ED07  9D 5B 02           STA LSTJOB,X
8316  ED0A  A9 01              LDA #1         ; PUT SAL IN BUFFER
8317  ED0C  20 F1 CF           JSR PUTBYT
8318  ED0F  A9 04              LDA #4         ; PUT SAH IN BUFFER
8319  ED11  20 F1 CF           JSR PUTBYT
8320  ED14  A9 01              LDA #1         ; INSERT PHONEY LINKS (0101)
8321  ED16  20 F1 CF           JSR PUTBYT
8322  ED19  20 F1 CF           JSR PUTBYT
8323  ED1C  AD 72 02           LDA NBTEMP
8324  ED1F  20 F1 CF           JSR PUTBYT     ; PUT IN DRVNUM
8325  ED22  A9 00              LDA #0
8326  ED24  20 F1 CF           JSR PUTBYT
8327  ED27  20 A7 ED           JSR MOVBUF     ; GET DISK NAME
8328  ED2A  20 93 DF           JSR GETACT
8329  ED2D  0A                 ASL A
8330  ED2E  AA                 TAX
8331  ED2F  D6 99              DEC BUFTAB,X
8332  ED31  D6 99              DEC BUFTAB,X
8333  ED33  A9 00              LDA #0         ; END OF THIS LINE
8334  ED35  20 F1 CF           JSR PUTBYT
8335  ED38  A9 01       DIR1   LDA #1          ; INSERT PHONEY LINKS ($0101)
8336  ED3A  20 F1 CF           JSR PUTBYT
8337  ED3D  20 F1 CF           JSR PUTBYT
8338  ED40  20 CE C6           JSR GETNAM     ; GET #BUFRS AND FILE NAME
8339  ED43  90 2C              BCC DIR3       ; TEST IF LAST ENTRY
8340  ED45  AD 72 02           LDA NBTEMP
8341  ED48  20 F1 CF           JSR PUTBYT
8342  ED4B  AD 73 02           LDA NBTEMP+1
8343  ED4E  20 F1 CF           JSR PUTBYT
8344  ED51  20 A7 ED           JSR MOVBUF
8345  ED54  A9 00              LDA #0         ; END OF ENTRY
8346  ED56  20 F1 CF           JSR PUTBYT
8347  ED59  D0 DD              BNE DIR1
8348  ED5B  20 93 DF    DIR10  JSR GETACT
8349  ED5E  0A                 ASL A
8350  ED5F  AA                 TAX
8351  ED60  A9 00              LDA #0
8352  ED62  95 99              STA BUFTAB,X
```

```
LINE# LOC   CODE        LINE


8353  ED64  A9 88              LDA #RDYTLK
8354  ED66  A4 82              LDY LINDX
8355  ED68  8D 54 02           STA DIRLST
8356  ED6B  99 F2 00           STA CHNRDY,Y   ; DIRECTORY LIST BUFFER FULL
8357  ED6E  A5 85              LDA DATA
8358  ED70  60                 RTS
8359  ED71              ;
8360  ED71              ;
8361  ED71  AD 72 02    DIR3   LDA NBTEMP      ; THIS IS END OF LOAD
8362  ED74  20 F1 CF           JSR PUTBYT
8363  ED77  AD 73 02           LDA NBTEMP+1
8364  ED7A  20 F1 CF           JSR PUTBYT
8365  ED7D  20 A7 ED           JSR MOVBUF
8366  ED80  20 93 DF           JSR GETACT
8367  ED83  0A                 ASL A
8368  ED84  AA                 TAX
8369  ED85  D6 99              DEC BUFTAB,X
8370  ED87  D6 99              DEC BUFTAB,X
8371  ED89  A9 00              LDA #0         ; END OF LISTING (000)
8372  ED8B  20 F1 CF           JSR PUTBYT
8373  ED8E  20 F1 CF           JSR PUTBYT
8374  ED91  20 F1 CF           JSR PUTBYT
8375  ED94  20 93 DF           JSR GETACT
8376  ED97  0A                 ASL A
8377  ED98  A8                 TAY
8378  ED99  B9 99 00           LDA BUFTAB,Y
8379  ED9C  A6 82              LDX LINDX
8380  ED9E  9D 44 02           STA LSTCHR,X
8381  EDA1  DE 44 02           DEC LSTCHR,X
8382  EDA4  4C 5B ED           JMP DIR10
8383  EDA7              ;
8384  EDA7              ;
8385  EDA7              ;
8386  EDA7              ;
8387  EDA7              ; TRANSFER FILE NAME TO LISTING BUFFER
8388  EDA7              ;
8389  EDA7  A0 00       MOVBUF LDY #0
8390  EDA9  B9 B1 02    MOVB1  LDA NAMBUF,Y
8391  EDAC  20 F1 CF           JSR PUTBYT
8392  EDAF  C8                 INY
8393  EDB0  C0 1B              CPY #27
8394  EDB2  D0 F5              BNE MOVB1
8395  EDB4  60                 RTS
8396  EDB5              ;
8397  EDB5              ;
8398  EDB5              ; GET CHAR FOR DIRECTORY LOADING
8399  EDB5              ;
8400  EDB5  20 37 D1    GETDIR JSR GETBYT
8401  EDB8  F0 01              BEQ GETD3
8402  EDBA  60                 RTS
8403  EDBB  85 85       GETD3  STA DATA
8404  EDBD  A4 82              LDY LINDX
8405  EDBF  B9 44 02           LDA LSTCHR,Y
8406  EDC2  F0 08              BEQ GD1
8407  EDC4  A9 80              LDA #EOIOUT
```

```
LINE# LOC   CODE        LINE


8408  EDC6  99 F2 00           STA CHNRDY,Y
8409  EDC9  A5 85              LDA DATA
8410  EDCB  60                 RTS
8411  EDCC              GD1
8412  EDCC  48                 PHA
8413  EDCD  20 38 ED           JSR DIR1
8414  EDD0  68                 PLA
8415  EDD1  60                 RTS
8416  EDD2              ;
8417  EDD2              ; .END
8417  EDD2              ;
8418  EDD2                     .LIB VERDIR
```

```
LINE# LOC   CODE        LINE


8420  EDD2              ; VALIDATE FILES WITH BAM
8421  EDD2              ; CREATE NEW BAM ACCORDING TO
8422  EDD2              ; CONTENTS OF FILES ENTERED IN DIR

8424  EDD2              VERDIR
8425  EDD2              VALDAT
8426  EDD2              ;VALIDATE IS SOFT-LOAD
8427  EDD2  20 D1 C1           JSR SIMPRS     ;EXTRACT DRIVE #
8428  EDD5  20 42 D0           JSR INITDR
8429  EDD8  A9 40              LDA #$40
8430  EDDA  8D F9 02           STA WBAM
8431  EDDD  20 05 EF           JSR NEWMPV     ;SET NEW BAM
8432  EDE0  A9 00              LDA #0
8433  EDE2  8D 92 02           STA DELIND
8434  EDE5  20 AC C5           JSR SRCHST     ;SEARCH FOR FIRST FILE
8435  EDE8  D0 3D              BNE VD25       ;FOUND ONE
8436  EDEA  A9 00       VD10   LDA #0          ; SET DIRECTORY SECTORS...
8437  EDEC  85 81              STA SECTOR     ;...IN BAM
8438  EDEE  AD D4 FE           LDA DIRTRK
8439  EDF1  85 80              STA TRACK
8440  EDF3  20 33 EE           JSR VMKBAM
8441  EDF6  A9 00              LDA #0
8442  EDF8  8D F9 02           STA WBAM
8443  EDFB  20 4D EF           JSR SCRBAM     ;WRITE OUT BAMS
8444  EDFE  4C 94 C1           JMP ENDCMD

8446  EE01  C8          VD15   INY
8447  EE02  B1 94              LDA (DIRBUF),Y
8448  EE04  48                 PHA            ;SAVE TRACK
8449  EE05  C8                 INY
8450  EE06  B1 94              LDA (DIRBUF),Y
8451  EE08  48                 PHA            ;SAVE SECTOR
8452  EE09  A0 13              LDY #19        ;GET SS TRACK
8453  EE0B  B1 94              LDA (DIRBUF),Y ; IS THIS RELATIVE ?
8454  EE0D  F0 0A              BEQ VD17       ;NO
8455  EE0F  85 80              STA TRACK      ;YES - SAVE TRACK
8456  EE11  C8                 INY
8457  EE12  B1 94              LDA (DIRBUF),Y ; GET SS SECTOR
8458  EE14  85 81              STA SECTOR
8459  EE16  20 33 EE           JSR VMKBAM     ;VALIDATE SS BY LINKS
8460  EE19  68          VD17   PLA
8461  EE1A  85 81              STA SECTOR     ;NOW DO DATA BLOCKS
8462  EE1C  68                 PLA
8463  EE1D  85 80              STA TRACK
8464  EE1F  20 33 EE           JSR VMKBAM     ;SET BIT USED IN BAM
8465  EE22  20 04 C6    VD20   JSR SRRE        ;SEARCH FOR MORE
8466  EE25  F0 C3              BEQ VD10       ;NO MORE FILES
8467  EE27              VD25
8468  EE27  A0 00              LDY #0
8469  EE29  B1 94              LDA (DIRBUF),Y
8470  EE2B  30 D4              BMI VD15
8471  EE2D  20 B6 C8           JSR DELDIR     ;NOT CLOSED DELETE DIR
8472  EE30  4C 22 EE           JMP VD20
8473  EE33              ;
8474  EE33              VMKBAM                 ;MARK BAM WITH FILE SECTORS
```

```
LINE# LOC   CODE        LINE


8475  EE33  20 5F D5           JSR TSCHK
8476  EE36  20 DE EF           JSR WUSED
8477  EE39  20 75 D4           JSR OPNIRD
8478  EE3C  A9 00       MRK2   LDA #0
8479  EE3E  20 C8 D4           JSR SETPNT
8480  EE41  20 37 D1           JSR GETBYT
8481  EE44  85 80              STA TRACK
8482  EE46  20 37 D1           JSR GETBYT
8483  EE49  85 81              STA SECTOR
8484  EE4B  A5 80              LDA TRACK
8485  EE4D  D0 03              BNE MRK1
8486  EE4F  4C 27 D2           JMP FRECHN
8487  EE52  20 DE EF    MRK1   JSR WUSED
8488  EE55  20 4D D4           JSR NXTBUF
8489  EE58  4C 3C EE           JMP MRK2
8490  EE5B              ;
8491  EE5B              ;.END
8491  EE5B              ;
8492  EE5B                     .LIB NEW
```

```
LINE# LOC   CODE         LINE


8494  EE5B               ;NEW: INITIALIZE A DISK, DISK IS
8495  EE5B               ; SOFT-SECTORED, BIT AVAIL. MAP,
8496  EE5B               ; DIRECTORY, & 1ST BLOCK ARE ALL INITED

8498  EE5B  20 12 C3    NEW    JSR ONEDRV
8499  EE5E  A5 E2              LDA FILDRV     ; SET UP DRIVE #
8500  EE60  10 05              BPL N101
8501  EE62  A9 33              LDA #BADFN     ; BAD DRIVE # GIVEN
8502  EE64  4C C8 C1           JMP CMDERR
8503  EE67  29 01       N101   AND #1
8504  EE69  85 7F              STA DRVNUM
8505  EE6B  20 00 C1           JSR SETLDS
8506  EE6E  A5 7F              LDA DRVNUM
8507  EE70  0A                 ASL A
8508  EE71  AA                 TAX
8509  EE72  AC 7B 02           LDY FILTBL+1   ; GET DISK ID
8510  EE75  CC 74 02           CPY CMDSIZ     ; ?IS THIS NEW OR CLEAR?
8511  EE78  F0 1A              BEQ N108       ; END OF CMD STRING
8512  EE7A  B9 00 02           LDA CMDBUF,Y   ; FORMAT DISK****
8513  EE7D  95 12              STA DSKID,X    ; STORE IN PROPER DRIVE
8514  EE7F  B9 01 02           LDA CMDBUF+1,Y ; (Y=0)
8515  EE82  95 13              STA DSKID+1,X
8516  EE84               ;
8517  EE84  20 07 D3           JSR CLRCHN     ; CLEAR ALL CHANNELS WHEN
                                                FORMATTING
8518  EE87  A9 01              LDA #1         ; ...IN TRACK, TRACK=1
8519  EE89  85 80              STA TRACK
8520  EE8B  20 C6 C8           JSR FORMAT     ;TRANSFER FORMAT TO RAM
8521  EE8E  20 53 F0           JSR CLRBAM     ; ZERO BAM
8522  EE91  4C A4 EE           JMP N110

8524  EE94  20 42 D0    N108   JSR INITDR      ; CLEAR DIRECTORY ONLY
8525  EE97  A6 7F              LDX DRVNUM
8526  EE99  BD 01 01           LDA DSKVER,X   ; USE CURRENT VERSION #
8527  EE9C  CD 24 FF           CMP VERNUM
8528  EE9F  F0 03              BEQ N110
8529  EEA1  4C 72 D5           JMP VNERR      ; WRONG VERSION #
8530  EEA4              N110
8531  EEA4  20 05 EF           JSR NEWMAP     ; NEW BAM

8533  EEA7  A5 F9              LDA JOBNUM
8534  EEA9  A8                 TAY
8535  EEAA  0A                 ASL A
8536  EEAB  AA                 TAX
8537  EEAC  AD D7 FE           LDA DSKNAM     ; SET PTR TO DISK NAME
8538  EEAF  95 99              STA BUFTAB,X
8539  EEB1  AE 7A 02           LDX FILTBL
8540  EEB4  A9 1B              LDA #27
8541  EEB6  20 6E C6           JSR TRNAME     ; TRANSFER CMD BUF TO BAM

8543  EEB9  A0 12              LDY #$12
8544  EEBB  A6 7F              LDX DRVNUM
8545  EEBD  AD 24 FF           LDA VERNUM     ; SET DOS'S CURRENT FORMAT TYPE
8546  EEC0  9D 01 01           STA DSKVER,X
8547  EEC3  8A                 TXA
8548  EEC4  0A                 ASL A
```

```
LINE# LOC   CODE        LINE


8549  EEC5  AA                 TAX
8550  EEC6  B5 12              LDA DSKID,X    ; WRITE DIRECTORY'S I.D.
8551  EEC8  91 94              STA (DIRBUF),Y
8552  EECA  C8                 INY
8553  EECB  B5 13              LDA DSKID+1,X
8554  EECD  91 94              STA (DIRBUF),Y

8556  EECF  C8                 INY
8557  EED0  C8                 INY
8558  EED1  A9 32              LDA #DOSVER+$30 ; WRITE DIRECTORY DOS VERSION
8559  EED3  91 94              STA (DIRBUF),Y
8560  EED5  C8                 INY
8561  EED6  AD 24 FF           LDA VERNUM     ; WRITE DIRECTORY FORMAT TYPE
8562  EED9  91 94              STA (DIRBUF),Y
8563  EEDB              ;
8564  EEDB  A0 02              LDY #2
8565  EEDD  91 6D              STA (BMPNT),Y  ; WRITE DISKETTE'S FORMAT TYPE
8566  EEDF  AD D4 FE           LDA DIRTRK
8567  EEE2  85 80              STA TRACK
8568  EEE4  20 E1 EF           JSR USEDTS     ; SET BAM BLOCK USED
8569  EEE7  A9 01              LDA #1
8570  EEE9  85 81              STA SECTOR
8571  EEEB  20 E1 EF           JSR USEDTS     ; SET 1ST DIR BLOCK USED
8572  EEEE  20 4D EF           JSR SCRBAM     ; SCRUB THE BAM
8573  EEF1  20 53 F0           JSR CLRBAM     ; SET TO ALL 0'S
8574  EEF4  A0 01              LDY #1
8575  EEF6  A9 FF              LDA #$FF       ; SET END LINK
8576  EEF8  91 6D              STA (BMPNT),Y
8577  EEFA  20 64 D4           JSR DRTWRT     ; CLEAR DIRECTORY
8578  EEFD  C6 81              DEC SECTOR
8579  EEFF  20 60 D4           JSR DRTRD      ; READ BAM BACK

8581  EF02  4C 94 C1           JMP ENDCMD

8583  EF05              ;.END
8583  EF05              ;
8584  EF05                     .LIB MAP
```

```
LINE# LOC   CODE         LINE


8587  EF05               ;
8588  EF05               ; BUILD A NEW MAP ON DISKETTE
8589  EF05               ;
8590  EF05               NEWMAP
8591  EF05               NEWMPV
8592  EF05  20 1F F1            JSR CLNBAM
8593  EF08  A0 00               LDY #0
8594  EF0A  A9 12               LDA #18         ; SET LINK TO 18.1
8595  EF0C  91 6D               STA (BMPNT),Y
8596  EF0E  C8                  INY
8597  EF0F  98                  TYA
8598  EF10  91 6D               STA (BMPNT),Y
8599  EF12  C8                  INY
8600  EF13  C8                  INY
8601  EF14  C8                  INY             ; .Y=4
8602  EF15               NM10
8603  EF15  A9 00               LDA #0          ; CLEAR TRACK MAP
8604  EF17  85 6F               STA T0
8605  EF19  85 70               STA T1
8606  EF1B  85 71               STA T2
8607  EF1D               ;
8608  EF1D  98                  TYA
8609  EF1E  4A                  LSR A
8610  EF1F  4A                  LSR A           ; .A=TRACK #
8611  EF20  20 99 F2            JSR MAXSEC
8612  EF23  91 6D               STA (BMPNT),Y
8613  EF25  C8                  INY
8614  EF26  AA                  TAX
8615  EF27               NM20
8616  EF27  38                  SEC             ; SET MAP BITS
8617  EF28  26 6F               ROL T0
8618  EF2A  26 70               ROL T1
8619  EF2C  26 71               ROL T2
8620  EF2E  CA                  DEX
8621  EF2F  D0 F6               BNE NM20
8622  EF31               NM30                    ; .X=0
8623  EF31  B5 6F               LDA T0,X
8624  EF33  91 6D               STA (BMPNT),Y
8625  EF35  C8                  INY
8626  EF36  E8                  INX
8627  EF37  E0 03               CPX #3
8628  EF39  90 F6               BCC NM30
8629  EF3B  C0 90               CPY #$90        ; END OF BAM
8630  EF3D  90 D6               BCC NM10
8631  EF3F  4C 75 D0            JMP NFCALC      ; CALC # FREE SECTORS
8632  EF42               ;
8633  EF42               ;
8634  EF42               ;
8635  EF42               ;
8636  EF42               ; WRITE OUT THE BIT MAP TO
8637  EF42               ; THE DRIVE IN LSTJOB(ACTIVE)
8638  EF42               ;
8639  EF42  20 93 DF     MAPOUT JSR GETACT
8640  EF45  AA                  TAX
```

```
LINE# LOC   CODE        LINE


8641  EF46  BD 5B 02           LDA LSTJOB,X
8642  EF49  29 01       MO10   AND #1
8643  EF4B  85 7F              STA DRVNUM     ; CHECK BAM BEFORE WRITING
8644  EF4D              ;
8645  EF4D              ; WRITE BAM ACCORDING TO DRVNUM
8646  EF4D              ;
8647  EF4D              SCRBAM
8648  EF4D  A4 7F              LDY DRVNUM
8649  EF4F  B9 51 02           LDA MDIRTY,Y
8650  EF52  D0 01              BNE SB10
8651  EF54  60                 RTS            ; NOT DIRTY
8652  EF55              SB10
8653  EF55  A9 00              LDA #0         ; SET TO CLEAN BAM
8654  EF57  99 51 02           STA MDIRTY,Y
8655  EF5A  20 88 EF           JSR SETBPT     ; SET BIT MAP PTR
8656  EF5D  A5 7F              LDA DRVNUM
8657  EF5F  0A                 ASL A
8658  EF60  48                 PHA
8659  EF61              ;PUT MEMORY IMAGES TO BAM
8660  EF61  20 F3 F0           JSR PUTBAM
8661  EF64  68                 PLA
8662  EF65  18                 CLC
8663  EF66  69 01              ADC #1
8664  EF68  20 F3 F0           JSR PUTBAM
8665  EF6B              ; VERIFY THE BAM BLOCK COUNT
8666  EF6B              ; MATCHES THE BITS
8667  EF6B              ;
8668  EF6B  A5 80              LDA TRACK
8669  EF6D  48                 PHA            ; SAVE TRACK VAR
8670  EF6E  A9 01              LDA #1
8671  EF70  85 80              STA TRACK
8672  EF72              SB20
8673  EF72  0A                 ASL A
8674  EF73  0A                 ASL A
8675  EF74  85 6D              STA BMPNT
8676  EF76  20 6E F2           JSR AVCK       ; CHECK AVAILABLE BLOCKS
8677  EF79  E6 80              INC TRACK
8678  EF7B  A5 80              LDA TRACK
8679  EF7D  CD 26 FF           CMP MAXTRK
8680  EF80  90 F0              BCC SB20
8681  EF82  68                 PLA            ; RESTORE TRACK VAR
8682  EF83  85 80              STA TRACK
8683  EF85  4C 8A D5           JMP DOWRIT     ; WRITE IT OUT
```

```
LINE# LOC   CODE        LINE


8685  EF88              ;
8686  EF88              ; SET BIT MAP PTR, READ IN BAM IF NEC.
8687  EF88              ;
8688  EF88              SETBPT
8689  EF88  20 5D F1           JSR BAM2A
8690  EF8B  AA                 TAX
8691  EF8C  20 2D F1           JSR REDBAM     ; READ BAM IF NOT IN
8692  EF8F  A6 F9              LDX JOBNUM
8693  EF91  BD 2F FF           LDA BUFIND,X   ; SET THE PTR
8694  EF94  85 6E              STA BMPNT+1
8695  EF96  A9 00              LDA #0
8696  EF98  85 6D              STA BMPNT
8697  EF9A  60                 RTS
8698  EF9B              ;
8699  EF9B              ; CALC THE NUMBER OF FREE BLOCKS ON DRVNUM
8700  EF9B              ;
8701  EF9B              NUMFRE
8702  EF9B  A6 7F              LDX DRVNUM
8703  EF9D  BD FA 02           LDA NDBL,X
8704  EFA0  8D 72 02           STA NBTEMP
8705  EFA3  BD FC 02           LDA NDBH,X
8706  EFA6  8D 73 02           STA NBTEMP+1
8707  EFA9  60                 RTS
8708  EFAA              ; .END
8708  EFAA              ;
8709  EFAA                     .LIB FRETS
```

```
LINE# LOC   CODE        LINE



8712  EFAA              ;MARK A TRACK,SECTOR AS FREE IN BAM
8713  EFAA              WFREE
8714  EFAA  20 3F F0           JSR FIXBAM
8715  EFAD              ;
8716  EFAD              FRETS
8717  EFAD  20 1D F0           JSR FREUSE     ;CALC INDEX INTO BAM
8718  EFB0              FRETS2
8719  EFB0  38                 SEC            ;FLAG FOR NO ACTION
8720  EFB1  D0 22              BNE FRERTS     ;FREE ALREADY
8721  EFB3  B1 6D              LDA (BMPNT),Y  ;NOT FREE, FREE IT
8722  EFB5  1D 37 F0           ORA BMASK,X
8723  EFB8  91 6D              STA (BMPNT),Y
8724  EFBA  20 D6 EF           JSR DTYBAM     ;SET DIRTY FLAG
8725  EFBD  A4 6F              LDY TEMP
8726  EFBF  18                 CLC
8727  EFC0  B1 6D              LDA (BMPNT),Y  ;ADD ONE
8728  EFC2  69 01              ADC #1
8729  EFC4  91 6D              STA (BMPNT),Y
8730  EFC6  A5 80              LDA TRACK
8731  EFC8  CD D4 FE           CMP DIRTRK
8732  EFCB  F0 3B              BEQ USE10
8733  EFCD              ;
8734  EFCD  FE FA 02           INC NDBL,X
8735  EFD0  D0 03              BNE FRE10
8736  EFD2  FE FC 02           INC NDBH,X
8737  EFD5              FRE10
8738  EFD5  60          FRERTS RTS
8739  EFD6              ;
8740  EFD6              DTYBAM
8741  EFD6  A6 7F              LDX DRVNUM
8742  EFD8  A9 01              LDA #1
8743  EFDA  9D 51 02           STA MDIRTY,X   ;SET DIRTY FLAG
8744  EFDD  60                 RTS
8745  EFDE              ;
8746  EFDE              ;MARK TRACK,SECTOR,(BMPNT) AS USED
8747  EFDE              ;
8748  EFDE              WUSED
8749  EFDE  20 3F F0           JSR FIXBAM
8750  EFE1              ;
8751  EFE1              USEDTS                  ;CALC INDEX INTO BAM
8752  EFE1  20 1D F0           JSR FREUSE
8753  EFE4  F0 36              BEQ USERTS     ;USED, NO ACTION
8754  EFE6  B1 6D              LDA (BMPNT),Y  ;GET BITS
8755  EFE8  5D 37 F0           EOR BMASK,X    ;MARK SEC USED
8756  EFEB  91 6D              STA (BMPNT),Y
8757  EFED  20 D6 EF           JSR DTYBAM
8758  EFF0  A4 6F              LDY TEMP
8759  EFF2  B1 6D              LDA (BMPNT),Y  ;GET COUNT
8760  EFF4  38                 SEC
8761  EFF5  E9 01              SBC #1         ; DEC ONE (C=0)
8762  EFF7  91 6D              STA (BMPNT),Y  ;SAVE IT
8763  EFF9  A5 80              LDA TRACK
8764  EFFB  CD D4 FE           CMP DIRTRK
8765  EFFE  F0 0B              BEQ USE20
```

```
LINE# LOC   CODE         LINE


8766  F000               ;
8767  F000  BD FA 02            LDA NDBL,X
8768  F003  D0 03               BNE USE10
8769  F005  DE FC 02            DEC NDBH,X
8770  F008               USE10
8771  F008  DE FA 02            DEC NDBL,X
8772  F00B               USE20
8773  F00B  BD FC 02            LDA NDBH,X
8774  F00E  D0 0C               BNE USERTS
8775  F010  BD FA 02            LDA NDBL,X
8776  F013  C9 03               CMP #3
8777  F015  B0 05               BCS USERTS
8778  F017  A9 72               LDA #DSKFUL
8779  F019  20 C7 E6            JSR ERRMSG
8780  F01C  60           USERTS RTS
8781  F01D               ;
8782  F01D               ;CALCULATES INDEX INTO BAM
8783  F01D               ;FOR FRETS AND USEDTS
8784  F01D               ;
8785  F01D               FREUSE
8786  F01D  20 5F F0            JSR SETBAM
8787  F020  98                  TYA
8788  F021               FREUS2
8789  F021  85 6F               STA TEMP        ;SAVE INDEX
8790  F023               FREUS3
8791  F023  A5 81               LDA SECTOR      ;A=SECTOR/8
8792  F025  4A                  LSR A
8793  F026  4A                  LSR A
8794  F027  4A                  LSR A           ;FOR WHICH OF THREE BYTES
8795  F028  38                  SEC
8796  F029  65 6F               ADC TEMP        ;CALC INDEX
8797  F02B  A8                  TAY
8798  F02C  A5 81               LDA SECTOR      ;BIT IN THAT BYTE
8799  F02E  29 07               AND #%00000111
8800  F030  AA                  TAX
8801  F031  B1 6D               LDA (BMPNT),Y   ;GET THE BYTE
8802  F033  3D 37 F0            AND BMASK,X     ;TEST IT
8803  F036  60                  RTS             ;Z=1=USED,Z=0=FREE
8804  F037               ;
8805  F037               ;
8806  F037  01           BMASK  .BYTE $01
8807  F038  02                  .BYTE $02
8808  F039  04                  .BYTE $04
8809  F03A  08                  .BYTE $08
8810  F03B  10                  .BYTE $10
8811  F03C  20                  .BYTE $20
8812  F03D  40                  .BYTE $40
8813  F03E  80                  .BYTE $80
8814  F03F               ;
8815  F03F               FIXBAM                  ;WRITE THE BAM ACCORDING TO WBAM
                                                 FLAG
8816  F03F  A9 FF               LDA #$FF
8817  F041  2C F9 02            BIT WBAM        ;IF BAM DIRTY
8818  F044  F0 0C               BEQ FBAM10      ;TEST FLAGS
8819  F046  10 0A               BPL FBAM10
8820  F048  70 08               BVS FBAM10
```

```
LINE# LOC   CODE        LINE


8821  F04A              ;
8822  F04A  A9 00              LDA #0
8823  F04C  8D F9 02           STA WBAM       ;CLEAR FLAG
8824  F04F  4C 8A D5           JMP DOWRIT     ;WRITE BAM TO DISK
8825  F052              ;
8826  F052              FBAM10
8827  F052  60                 RTS            ;ELSE EXIT
8828  F053              ;
8829  F053              ;
8830  F053              ;CLEAR BAM BUFFER
8831  F053              CLRBAM
8832  F053  20 88 EF           JSR SETBPT     ;GET POINTER TO BAM
8833  F056  A0 00              LDY #0
8834  F058  98                 TYA
8835  F059              CLB1
8836  F059  91 6D              STA (BMPNT),Y
8837  F05B  C8                 INY
8838  F05C  D0 FB              BNE CLB1
8839  F05E  60                 RTS
8840  F05F              ;
8841  F05F              SETBAM                         ;SET BAM IMAGE IN MEMORY
8842  F05F  A5 6F              LDA T0         ;SAVE TEMPS
8843  F061  48                 PHA
8844  F062  A5 70              LDA T1
8845  F064  48                 PHA
8846  F065  A6 7F              LDX DRVNUM
8847  F067  B5 FF              LDA NODRV,X
8848  F069  F0 05              BEQ SBM10
8849  F06B              ;
8850  F06B  A9 74              LDA #NODRIV    ;NO DRIVE
8851  F06D  20 48 E6           JSR CMDER3
8852  F070              SBM10
8853  F070  20 5D F1           JSR BAM2A
8854  F073  85 6F              STA T0         ;T0:= INDEX INTO BUF0
8855  F075  8A                 TXA
8856  F076  0A                 ASL A
8857  F077  85 70              STA T1         ;T1:= 2*DRVNUM
8858  F079  AA                 TAX
8859  F07A  A5 80              LDA TRACK
8860  F07C  DD 9D 02           CMP TBAM,X     ;CHECK BAM TABLE FOR TRACK
8861  F07F  F0 0B              BEQ SBM30      ;IT'S IN
8862  F081              ;
8863  F081  E8                 INX
8864  F082  86 70              STX T1         ;CHECK NEXT ENTRY
8865  F084  DD 9D 02           CMP TBAM,X
8866  F087  F0 03              BEQ SBM30      ;IT'S IN
8867  F089              ;
8868  F089  20 A9 F0           JSR SWAP       ;SWAP TRACK BAM IN TABLE
8869  F08C              SBM30
8870  F08C  A5 70              LDA T1
8871  F08E  A6 7F              LDX DRVNUM
8872  F090  9D 9B 02           STA UBAM,X     ;SET LAST USED PTR
8873  F093  0A                 ASL A
8874  F094  0A                 ASL A
8875  F095  18                 CLC
```

```
LINE# LOC   CODE        LINE


8876  F096  69 A1              ADC #<BAM      ;SET ACTUAL PTR
8877  F098  85 6D              STA BMPNT
8878  F09A  A9 02              LDA #>BAM
8879  F09C  69 00              ADC #0
8880  F09E  85 6E              STA BMPNT+1
8881  F0A0  A0 00              LDY #0
8882  F0A2  68                 PLA
8883  F0A3  85 70              STA T1
8884  F0A5  68                 PLA
8885  F0A6  85 6F              STA T0
8886  F0A8  60                 RTS
8887  F0A9              ;
8888  F0A9              ;
8889  F0A9              SWAP                           ;SWAP IMAGES OF BAM
8890  F0A9  A6 6F              LDX T0         ;GET CHANNEL NUMBER
8891  F0AB  20 2D F1           JSR REDBAM     ;READ BAM IF NOT IN
8892  F0AE  A5 7F              LDA DRVNUM
8893  F0B0  AA                 TAX
8894  F0B1  0A                 ASL A
8895  F0B2  1D 9B 02           ORA UBAM,X     ;SWAP OUT LEAST USED IMAGE
8896  F0B5  49 01              EOR #1
8897  F0B7  29 03              AND #3
8898  F0B9  85 70              STA T1
8899  F0BB  20 F3 F0           JSR PUTBAM     ;PUT TO BAM
8900  F0BE  A5 F9              LDA JOBNUM
8901  F0C0  0A                 ASL A
8902  F0C1  AA                 TAX
8903  F0C2  A5 80              LDA TRACK
8904  F0C4  0A                 ASL A
8905  F0C5  0A                 ASL A
8906  F0C6  95 99              STA BUFTAB,X   ;SET PTR
8907  F0C8  A5 70              LDA T1
8908  F0CA  0A                 ASL A
8909  F0CB  0A                 ASL A
8910  F0CC  A8                 TAY
8911  F0CD              SWAP3                          ;TRANSFER BAM TO MEM IMAGE
8912  F0CD  A1 99              LDA (BUFTAB,X)
8913  F0CF  99 A1 02           STA BAM,Y
8914  F0D2  A9 00              LDA #0
8915  F0D4  81 99              STA (BUFTAB,X) ;CLEAR BAM
8916  F0D6  F6 99              INC BUFTAB,X
8917  F0D8  C8                 INY
8918  F0D9  98                 TYA
8919  F0DA  29 03              AND #%00000011
8920  F0DC  D0 EF              BNE SWAP3
8921  F0DE              ;
8922  F0DE  A6 70              LDX T1
8923  F0E0  A5 80              LDA TRACK
8924  F0E2  9D 9D 02           STA TBAM,X     ;SET TRACK # FOR IMAGE
8925  F0E5              ;
8926  F0E5  AD F9 02           LDA WBAM
8927  F0E8  D0 03              BNE SWAP4      ;DON'T WRITE NOW
8928  F0EA  4C 8A D5           JMP DOWRIT
8929  F0ED              SWAP4
8930  F0ED  09 80              ORA #%10000000 ;SET PENDING WRITE FLAG
```

```
LINE# LOC   CODE         LINE


8931  F0EF  8D F9 02            STA WBAM
8932  F0F2  60                  RTS
8933  F0F3               ;
8934  F0F3               PUTBAM                     ;PUT MEM IMAGE TO BAM
8935  F0F3  A8                  TAY
8936  F0F4  B9 9D 02            LDA TBAM,Y
8937  F0F7  F0 25               BEQ SWAP2      ;NO IMAGE HERE
8938  F0F9  48                  PHA            ;SAVE TRACK #
8939  F0FA  A9 00               LDA #0
8940  F0FC  99 9D 02            STA TBAM,Y     ;CLEAR TRACK FLAG
8941  F0FF  A5 F9               LDA JOBNUM
8942  F101  0A                  ASL A
8943  F102  AA                  TAX
8944  F103  68                  PLA
8945  F104  0A                  ASL A
8946  F105  0A                  ASL A
8947  F106  95 99               STA BUFTAB,X   ;SET PTR IN BAM
8948  F108  98                  TYA
8949  F109  0A                  ASL A
8950  F10A  0A                  ASL A
8951  F10B  A8                  TAY
8952  F10C               SWAP1                      ;TRANSFER IMAGE TO BAM
8953  F10C  B9 A1 02            LDA BAM,Y
8954  F10F  81 99               STA (BUFTAB,X)
8955  F111  A9 00               LDA #0
8956  F113  99 A1 02            STA BAM,Y      ;CLEAR IMAGE
8957  F116  F6 99               INC BUFTAB,X
8958  F118  C8                  INY
8959  F119  98                  TYA
8960  F11A  29 03               AND #%00000011
8961  F11C  D0 EE               BNE SWAP1
8962  F11E               SWAP2
8963  F11E  60                  RTS
8964  F11F               ;
8965  F11F               ;
8966  F11F               CLNBAM                     ;CLEAN TRACK # FOR IMAGES
8967  F11F  A5 7F               LDA DRVNUM
8968  F121  0A                  ASL A
8969  F122  AA                  TAX
8970  F123  A9 00               LDA #0
8971  F125  9D 9D 02            STA TBAM,X
8972  F128  E8                  INX
8973  F129  9D 9D 02            STA TBAM,X
8974  F12C  60                  RTS
8975  F12D               ;
8976  F12D               ;
8977  F12D               ;
8978  F12D               REDBAM                     ;READ IN BAM IF NOT PRESENT
8979  F12D  B5 A7               LDA BUF0,X
8980  F12F  C9 FF               CMP #$FF
8981  F131  D0 25               BNE RBM20      ;IT IS IN MEMORY
8982  F133  8A                  TXA
8983  F134  48                  PHA            ;SAVE CHANNEL PTR
8984  F135  20 8E D2            JSR GETBUF     ;GO FIND A BUFFER
8985  F138  AA                  TAX
```

```
LINE# LOC   CODE        LINE


8986  F139  10 05              BPL RBM10
8987  F13B              ;
8988  F13B  A9 70              LDA #NOCHNL    ;NO BUFFERS AROUND
8989  F13D  20 C8 C1           JSR CMDERR
8990  F140              RBM10
8991  F140  86 F9              STX JOBNUM     ;SAVE JOBNUM ASSIGNED
8992  F142  68                 PLA
8993  F143  A8                 TAY
8994  F144  8A                 TXA
8995  F145  09 80              ORA #$80
8996  F147  99 A7 00           STA BUF0,Y     ;SET AS INACTIVE FOR STEALING
8997  F14A              ;READ IN BAM
8998  F14A  0A                 ASL A
8999  F14B  AA                 TAX
9000  F14C  AD D4 FE           LDA DIRTRK
9001  F14F  95 06              STA HDRS,X
9002  F151  A9 00              LDA #0
9003  F153  95 07              STA HDRS+1,X
9004  F155  4C 86 D5           JMP DOREAD
9005  F158              RBM20
9006  F158  29 0F              AND #%00001111 ;SET BAM'S JOBNUM
9007  F15A  85 F9              STA JOBNUM
9008  F15C  60                 RTS
9009  F15D              ;
9010  F15D              ;SET BAM POINTER IN BUF0/1 TABLES
9011  F15D              ;
9012  F15D              BAM2A                  ;LEAVE IN .A
9013  F15D  A9 06              LDA #BLINDX
9014  F15F  A6 7F              LDX DRVNUM
9015  F161  D0 03              BNE B2X10
9016  F163              ;
9017  F163  18                 CLC
9018  F164  69 07              ADC #MXCHNS+1
9019  F166              B2X10
9020  F166  60                 RTS
9021  F167              ;
9022  F167              BAM2X                  ;LEAVE IN .X
9023  F167  20 5D F1           JSR BAM2A
9024  F16A  AA                 TAX
9025  F16B  60                 RTS
9026  F16C              ;
9027  F16C              ;
9028  F16C              ; .END
9028  F16C              ;
9029  F16C                     .LIB TSTFND
```

```
LINE# LOC   CODE        LINE


9031  F16C              ; NEXT TRACK & SECTOR
9032  F16C              ; RETURNS NEXT AVAILABLE TRACK & SECTOR
9033  F16C              ; GIVEN CURRENT T & S
9034  F16C              ;
9035  F16C              ; ALLOCATION IS FROM TRACK 18
9036  F16C              ; TOWARDS 1 & 35, BY FULL TRACKS
9037  F16C              NXTTS
9038  F16C  20 3E DE           JSR GETHDR
9039  F16F  A9 03              LDA #3
9040  F171  85 6F              STA TEMP
9041  F173  A9 01              LDA #1          ;SET NO WRITE BAM
9042  F175  0D F9 02           ORA WBAM
9043  F178  8D F9 02           STA WBAM
9044  F17B              NXTDS
9045  F17B              NXT1
9046  F17B  A5 6F              LDA TEMP
9047  F17D  48                 PHA             ;SAVE TEMP
9048  F17E  20 5F F0           JSR SETBAM
9049  F181  68                 PLA
9050  F182  85 6F              STA TEMP        ;RESTORE TEMP
9051  F184  B1 6D              LDA (BMPNT),Y
9052  F186  D0 39              BNE FNDNXT
9053  F188  A5 80              LDA TRACK
9054  F18A  CD D4 FE           CMP DIRTRK
9055  F18D  F0 19              BEQ NXTERR
9056  F18F  90 1C              BCC NXT2
9057  F191  E6 80              INC TRACK
9058  F193  A5 80              LDA TRACK
9059  F195  CD 26 FF           CMP MAXTRK
9060  F198  D0 E1              BNE NXT1
9061  F19A  AE D4 FE           LDX DIRTRK
9062  F19D  CA                 DEX
9063  F19E  86 80              STX TRACK
9064  F1A0  A9 00              LDA #0
9065  F1A2  85 81              STA SECTOR
9066  F1A4  C6 6F              DEC TEMP
9067  F1A6  D0 D3              BNE NXT1
9068  F1A8  A9 72       NXTERR LDA #DSKFUL
9069  F1AA  20 C8 C1           JSR CMDERR
9070  F1AD  C6 80       NXT2   DEC TRACK
9071  F1AF  D0 CA              BNE NXT1
9072  F1B1  AE D4 FE           LDX DIRTRK
9073  F1B4  E8                 INX
9074  F1B5  86 80              STX TRACK
9075  F1B7  A9 00              LDA #0
9076  F1B9  85 81              STA SECTOR
9077  F1BB  C6 6F              DEC TEMP
9078  F1BD  D0 BC              BNE NXT1
9079  F1BF  F0 E7              BEQ NXTERR      ;BRANCH ALWAYS
9080  F1C1              ;
9081  F1C1              ; FIND THE NEXT OPTIMUM SECTOR
9082  F1C1              ; NEXT SECTOR=CURRENT SECTOR+N
9083  F1C1              ;
9084  F1C1  A5 81       FNDNXT LDA SECTOR
9085  F1C3  18                 CLC
```

```
LINE# LOC   CODE        LINE


9086  F1C4  65 69              ADC SECINC
9087  F1C6  85 81              STA SECTOR
9088  F1C8  A5 80              LDA TRACK
9089  F1CA  20 99 F2           JSR MAXSEC
9090  F1CD  8D 4E 02           STA LSTSEC
9091  F1D0  8D 4D 02           STA CMD
9092  F1D3  C5 81              CMP SECTOR
9093  F1D5  B0 0C              BCS FNDN0
9094  F1D7              ;
9095  F1D7  38                 SEC
9096  F1D8  A5 81              LDA SECTOR
9097  F1DA  ED 4E 02           SBC LSTSEC
9098  F1DD  85 81              STA SECTOR
9099  F1DF  F0 02              BEQ FNDN0
9100  F1E1              ;
9101  F1E1  C6 81              DEC SECTOR
9102  F1E3              FNDN0
9103  F1E3  20 48 F2           JSR GETSEC
9104  F1E6  F0 03              BEQ FNDN2
9105  F1E8              FNDN1
9106  F1E8  4C DE EF           JMP WUSED
9107  F1EB              FNDN2
9108  F1EB  A9 00              LDA #0
9109  F1ED  85 81              STA SECTOR
9110  F1EF  20 48 F2           JSR GETSEC
9111  F1F2  D0 F4              BNE FNDN1
9112  F1F4  4C 43 F2           JMP DERR
9113  F1F7              ;
9114  F1F7              ;
9115  F1F7              ; RETURNS OPTIMUM INITIAL TRACK,SECTOR
9116  F1F7              ;
9117  F1F7              INTTS
9118  F1F7  A9 01              LDA #1
9119  F1F9  0D F9 02           ORA WBAM
9120  F1FC  8D F9 02           STA WBAM
9121  F1FF  A5 86              LDA R0
9122  F201  48                 PHA            ;SAVE TEMP VAR
9123  F202              ;R0:= 1
9124  F202  A9 01              LDA #1
9125  F204  85 86              STA R0
9126  F206              ITS1                   ;TRACK:= DIRTRK-R0
9127  F206  AD D4 FE           LDA DIRTRK
9128  F209  38                 SEC
9129  F20A  E5 86              SBC R0
9130  F20C  85 80              STA TRACK
9131  F20E              ;IF T>0
9132  F20E  90 09              BCC ITS2
9133  F210  F0 07              BEQ ITS2
9134  F212              ;THEN BEGIN
9135  F212  20 5F F0           JSR SETBAM     ;SET THE BAM POINTER
9136  F215              ;IF @B..Y. THEN GOTO FNDSEC
9137  F215  B1 6D              LDA (BMPNT),Y
9138  F217  D0 1B              BNE FNDSEC
9139  F219              ;END
9140  F219              ITS2                   ;TRACK:= DIRTRK+R0
```

```
LINE# LOC   CODE         LINE


9141  F219  AD D4 FE            LDA DIRTRK
9142  F21C  18                  CLC
9143  F21D  65 86               ADC R0
9144  F21F  85 80               STA TRACK
9145  F221               ;R0:= R0+1
9146  F221  E6 86               INC R0
9147  F223               ;IF TRACK >=MAXTRK THEN CMDER2(SYSTS)
9148  F223  CD 26 FF            CMP MAXTRK
9149  F226  90 05               BCC ITS3
9150  F228               ;
9151  F228  A9 67               LDA #SYSTS
9152  F22A  20 45 E6            JSR CMDER2
9153  F22D               ITS3
9154  F22D  20 5F F0            JSR SETBAM      ;SET PTR
9155  F230               ;IF @B..Y.=0 THEN GOTO ITS1
9156  F230  B1 6D               LDA (BMPNT),Y
9157  F232  F0 D2               BEQ ITS1
9158  F234               FNDSEC
9159  F234  68                  PLA
9160  F235  85 86               STA R0          ; RESTORE R0
9161  F237  A9 00               LDA #0
9162  F239  85 81               STA SECTOR
9163  F23B  20 48 F2            JSR GETSEC
9164  F23E  F0 03               BEQ FND2
9165  F240  4C DE EF            JMP WUSED
9166  F243               ;
9167  F243               FND2
9168  F243               DERR
9169  F243  A9 71               LDA #DIRERR
9170  F245  20 45 E6            JSR CMDER2
9171  F248               ;
9172  F248               ;
9173  F248               ; SET BAM AND FIND AVAILABLE SECTOR
9174  F248               ; STARTING AT SECTOR
9175  F248               ;
9176  F248               GETSEC
9177  F248  20 5F F0            JSR SETBAM
9178  F24B  98                  TYA
9179  F24C  48                  PHA             ;SAVE .Y
9180  F24D  20 6E F2            JSR AVCK        ;CHECK BITS & COUNT
9181  F250               ;
9182  F250  A5 80               LDA TRACK
9183  F252  20 99 F2            JSR MAXSEC
9184  F255  8D 4E 02            STA LSTSEC      ;SAVE MAX SECTOR #
9185  F258  68                  PLA
9186  F259  85 6F               STA TEMP        ;TEMP:= OLD .Y FOR FREUS3
9187  F25B               GS10
9188  F25B  A5 81               LDA SECTOR
9189  F25D  CD 4E 02            CMP LSTSEC
9190  F260  B0 09               BCS GS20
9191  F262               ;
9192  F262  20 23 F0            JSR FREUS3
9193  F265  D0 06               BNE GS30
9194  F267               ;
9195  F267  E6 81               INC SECTOR
```

```
LINE# LOC   CODE        LINE


9196  F269  D0 F0              BNE GS10       ;BRA
9197  F26B              GS20
9198  F26B  A9 00              LDA #0
9199  F26D              GS30
9200  F26D  60                 RTS            ;(Z=1): USED

9202  F26E              ;BIT MAP VALIDITY CHECK
9203  F26E              AVCK
9204  F26E  A5 6F              LDA TEMP
9205  F270  48                 PHA            ;SAVE TEMP
9206  F271  A9 00              LDA #0
9207  F273  85 6F              STA TEMP
9208  F275              ;FOR .Y:=BAMSIZ TO 1 DO;
9209  F275  AC D5 FE           LDY BAMSIZ
9210  F278  88                 DEY
9211  F279              AC10                   ;FOR .X:=7 TO 0 DO;
9212  F279  A2 07              LDX #7         ;COUNT THE BITS
9213  F27B              AC20                   ;IF @B..Y. & BMASK.X.
9214  F27B              ; THEN TEMP:=TEMP+1
9215  F27B  B1 6D              LDA (BMPNT),Y
9216  F27D  3D 37 F0           AND BMASK,X
9217  F280  F0 02              BEQ AC30
9218  F282  E6 6F              INC TEMP
9219  F284              AC30                   ;END .X
9220  F284  CA                 DEX
9221  F285  10 F4              BPL AC20
9222  F287              ;END .Y
9223  F287  88                 DEY
9224  F288  D0 EF              BNE AC10
9225  F28A              ;IF @B..Y. <> TEMP
9226  F28A              ; THEN CMDER2(DIRERR)
9227  F28A  B1 6D              LDA (BMPNT),Y
9228  F28C  C5 6F              CMP TEMP
9229  F28E  D0 04              BNE AC40       ;COUNTS DO NOT MATCH
9230  F290              ;
9231  F290  68                 PLA
9232  F291  85 6F              STA TEMP       ;RESTORE TEMP
9233  F293  60                 RTS
9234  F294              AC40
9235  F294  A9 71              LDA #DIRERR
9236  F296  20 45 E6           JSR CMDER2

9238  F299              ; .A=TRACK # ,RETURNS #SECTORS ON THIS TRACK
9239  F299  AE 25 FF    MAXSEC LDX NZONES
9240  F29C  DD 25 FF    MAX1   CMP TRKNUM-1,X
9241  F29F  CA                 DEX
9242  F2A0  B0 FA              BCS MAX1
9243  F2A2  BD 20 FF           LDA NUMSEC,X
9244  F2A5  60                 RTS
9245  F2A6              ;
9246  F2A6              ;.END
9246  F2A6              ;
9247  F2A6                     .LIB SYSTEM
```

```
LINE# LOC   CODE         LINE


9249  F2A6               KILLP                  ; KILL PROTECTION
9250  F2A6               ; PHA
9251  F2A6               ; LDA #1
9252  F2A6               ; STA CFLG2 ;TELL CONTOLLER
9253  F2A6               ;KILLP2
9254  F2A6               ; LDA CFLG2 ;WAIT UNTIL HE'S GOT IT
9255  F2A6               ; BNE KILLP2
9256  F2A6               ;
9257  F2A6               ; PLA
9258  F2A6  60                  RTS
9259  F2A7               ;
9260  F2A7               ;.END
9260  F2A7               ;
```

```
LINE# LOC   CODE        LINE

9262  F2A7                     .LIB LCCIO
```

```
LINE# LOC   CODE         LINE


9264  F2A7               ;
9265  F2A7               CNTST= *
9266  F2A7               ;
9267  F2A7               ;
9268  F2A7               ;
9269  F2A7               ; DEFS FOR LOW COST CONTROLLER
9270  F2A7               ;
9271  F2A7               ;
9272  F2A7               ; WRITTEN BY GLENN STARK
9273  F2A7               ; 4/1/80
9274  F2A7               ;
9275  F2A7               ;
9276  F2A7               ; (C) COMMODORE BUSINESS MACHINES
9277  F2A7               ;
9278  F2A7               TIMER1 =$1805          ; TIMER 1 COUNTER
9279  F2A7               ;
9280  F2A7               ;
9281  F2A7               ;
9282  F2A7               ; MOS 6522
9283  F2A7               ; ADDRESS $1C00
9284  F2A7               ;
9285  F2A7                      * =$1C00
9286  1C00               ;
9287  1C00               DSKCNT *=*+1           ; PORT B
9288  1C01               ; DISK I/O CONTROL LINES
9289  1C01               ; BIT 0: STEP IN
9290  1C01               ; BIT 1: STEP OUT
9291  1C01               ; BIT 2: -MOTOR ON
9292  1C01               ; BIT 3: ACT LED
9293  1C01               ; BIT 4: WRITE PROTECT SENSE
9294  1C01               ; BIT 5: DENSITY SELECT 0
9295  1C01               ; BIT 6: DENSITY SELECT 1
9296  1C01               ; BIT 7: SYNC DETECT
9297  1C01               ;
9298  1C01               ;
9299  1C01               DATA2  *=*+1           ; PORT A
9300  1C02               ; GCR DATA INPUT AND OUTPUT PORT
9301  1C02               ;
9302  1C02               DDRB2  *=*+1           ; DATA DIRECTION CONTROL
9303  1C03               DDRA2  *=*+1           ; DATA DIRECTION CONTROL
9304  1C04               ;
9305  1C04               T1LC2  *=*+1           ; TIMER 1 LOW COUNTER
9306  1C05               T1HC2  *=*+1           ; TIMER 1 HI COUNTR
9307  1C06               ;
9308  1C06               T1LL2  *=*+1           ; TIMER 1 LOW LATCH
9309  1C07               T1HL2  *=*+1           ; TIMER 1 HI LATCH
9310  1C08               ;
9311  1C08               T2LL2  *=*+1           ; TIMER TWO LOW LATCH
9312  1C09               T2LH2  *=*+1           ; TIMER TWO HI LATCH
9313  1C0A               ;
9314  1C0A               SR2    *=*+1           ; SHIFT REGISTER
9315  1C0B               ;
9316  1C0B               ACR2   *=*+1
9317  1C0C               ;
9318  1C0C               PCR2   *=*+1
```

```
LINE# LOC   CODE         LINE


9319  1C0D               ;
9320  1C0D               IFR2   *=*+1
9321  1C0E               ;
9322  1C0E               IER2   *=*+1
9323  1C0F               ;
9324  1C0F               ;
9325  1C0F                      * =CNTST
9326  F2A7               ;
9327  F2A7               ;
9328  F2A7               ;
9329  F2A7               ;.END
9329  F2A7               ;
9330  F2A7                      .LIB LCCINIT
```

```
LINE# LOC   CODE        LINE


9332  F2A7              ;
9333  F2A7              ;
9334  F2A7              ;
9335  F2A7              ; INITIALIZATION OF CONTROLLER
9336  F2A7              ;
9337  F2A7              ;
9338  F2A7              ;
9339  F2A7  A9 6F       CNTINT LDA #%01101111 DATA DIRECTION
9340  F2A9  8D 02 1C           STA DDRB2
9341  F2AC  29 F0              AND #$FF-$08-$04-$03 ; TURN MOTOR OFF,SET PHASE
                                                A, LED OFF
9342  F2AE  8D 00 1C           STA DSKCNT
9343  F2B1              ;
9344  F2B1              ;
9345  F2B1  AD 0C 1C           LDA PCR2       ; SET EDGE AND LATCH MODE
9346  F2B4  29 FE              AND #$FF-$01   ; NEG EDGE PLEASE
9347  F2B6              ;
9348  F2B6              ;
9349  F2B6              ; CA2: SOE OUTPUT HI DISABLE S.O. INTO 6502
9350  F2B6              ;
9351  F2B6  09 0E              ORA #$0E
9352  F2B8              ;
9353  F2B8              ;
9354  F2B8              ; CB1 INPUT ONLY
9355  F2B8              ;
9356  F2B8              ; CB2 MODE CONTROL R/W
9357  F2B8              ;
9358  F2B8  09 E0              ORA #$E0
9359  F2BA  8D 0C 1C           STA PCR2
9360  F2BD              ;
9361  F2BD              ;
9362  F2BD  A9 41              LDA #$41       ; CONT IRQ, LATCH MODE
9363  F2BF  8D 0B 1C           STA ACR2
9364  F2C2  A9 00              LDA #0
9365  F2C4  8D 06 1C           STA T1LL2
9366  F2C7  A9 3A              LDA #TIM       ; / 8 MS /IRQ
9367  F2C9  8D 07 1C           STA T1HL2
9368  F2CC  8D 05 1C           STA T1HC2      ; GET 6522'S ATTENTION
9369  F2CF  A9 7F              LDA #$7F       ; CLEAR ALL IRQ SOURCES
9370  F2D1  8D 0E 1C           STA IER2
9371  F2D4              ;
9372  F2D4  A9 C0              LDA #$80+$40
9373  F2D6  8D 0D 1C           STA IFR2       ; CLEAR BIT
9374  F2D9  8D 0E 1C           STA IER2       ; ENABLE IRQ
9375  F2DC              ;
9376  F2DC              ;
9377  F2DC  A9 FF              LDA #$FF       ; NO CURRENT DRIVE
9378  F2DE  85 3E              STA CDRIVE
9379  F2E0  85 51              STA FTNUM      ; INIT FORMAT FLAG
9380  F2E2              ;
9381  F2E2  A9 08              LDA #$08       ; HEADER BLOCK ID
9382  F2E4  85 39              STA HBID
9383  F2E6              ;
9384  F2E6  A9 07              LDA #$07       ; DATA BLOCK ID
9385  F2E8  85 47              STA DBID
9386  F2EA              ;
```

```
LINE# LOC   CODE         LINE


9387  F2EA  A9 53               LDA #<INACT
9388  F2EC  85 62               STA NXTST
9389  F2EE  A9 FA               LDA #>INACT
9390  F2F0  85 63               STA NXTST+1
9391  F2F2               ;
9392  F2F2  A9 C8               LDA #200
9393  F2F4  85 64               STA MINSTP
9394  F2F6               ;
9395  F2F6  A9 04               LDA #4
9396  F2F8  85 5E               STA AS
9397  F2FA               ;
9398  F2FA  A9 04               LDA #$4
9399  F2FC  85 5F               STA AF
9400  F2FE               ;
9401  F2FE               ;.END
9401  F2FE               ;
9402  F2FE                      .LIB LCCCNTRL
```

```
LINE# LOC   CODE        LINE


9404  F2FE              ;
9405  F2FE              ;
9406  F2FE              ;
9407  F2FE              ; *CONTRL
9408  F2FE              ;
9409  F2FE              ; MAIN CONTROLLER LOOP
9410  F2FE              ;
9411  F2FE              ; SCANS JOB QUE FOR JOBS
9412  F2FE              ;
9413  F2FE              ; FINDS JOB ON CURRENT TRACK
9414  F2FE              ; IF IT EXISTS
9415  F2FE              ;
9416  F2FE              LCC
9417  F2FE              ;
9418  F2FE  BA                 TSX             ; SAVE CURRENT STACK POINTER
9419  F2FF  86 49              STX SAVSP
9420  F301              ;
9421  F301  AD 04 1C           LDA T1LC2       ; RESET IRQ FLAG
9422  F304              ;
9423  F304  AD 0C 1C           LDA PCR2        ; ENABLE S.O. TO 6502
9424  F307  09 0E              ORA #$0E        ; HI OUTPUT
9425  F309  8D 0C 1C           STA PCR2
9426  F30C              ;
9427  F30C              ;
9428  F30C              ;
9429  F30C  A0 05       TOP    LDY #NUMJOB-1    ; POINTER INTO JOB QUE
9430  F30E              ;
9431  F30E              CONT10
9432  F30E  B9 00 00           LDA JOBS,Y      ; FIND A JOB (MSB SET)
9433  F311  10 2E              BPL CONT20      ; NOT ONE HERE
9434  F313              ;
9435  F313  C9 D0              CMP #JUMPC      ; TEST IF ITS A JUMP COMMAND
9436  F315  D0 04              BNE CONT30
9437  F317              ;
9438  F317  98                 TYA             ; PUT JOB NUM IN .A
9439  F318  4C BE F3           JMP EX2
9440  F31B              ;
9441  F31B              ;
9442  F31B              CONT30
9443  F31B  29 01              AND #1          ; GET DRIVE #
9444  F31D  F0 07              BEQ CONT35
9445  F31F              ;
9446  F31F  84 3F              STY JOBN
9447  F321  A9 0F              LDA #$0F        ; BAD DRIVE # ERROR
9448  F323  4C B7 F9           JMP ERRR
9449  F326              ;
9450  F326  AA          CONT35 TAX
9451  F327  85 3D              STA DRIVE
9452  F329              ;
9453  F329  C5 3E              CMP CDRIVE      ; TEST IF CURRENT DRIVE
9454  F32B  F0 0A              BEQ CONT40
9455  F32D              ;
9456  F32D  20 CC F9           JSR TURNON      ; TURN ON DRIVE
9457  F330  A5 3D              LDA DRIVE
9458  F332  85 3E              STA CDRIVE
```

```
LINE# LOC   CODE        LINE


9459  F334  4C EA F9           JMP END        ; GO CLEAN UP
9460  F337              ;
9461  F337              ;
9462  F337  A5 20       CONT40 LDA DRVST       ; TEST IF MOTOR UP TO SPEED
9463  F339  30 03              BMI CONT50
9464  F33B              ;
9465  F33B  0A                 ASL A          ; TEST IF STEPPING
9466  F33C  10 09              BPL QUE        ; NOT STEPPING
9467  F33E              ;
9468  F33E  4C EA F9    CONT50 JMP END
9469  F341              ;
9470  F341  88          CONT20 DEY
9471  F342  10 CA              BPL CONT10
9472  F344              ;
9473  F344  4C EA F9           JMP END
9474  F347              ;
9475  F347              ;
9476  F347              ;
9477  F347              ;
9478  F347  A9 20       QUE    LDA #$20        ; STATUS=RUNNING
9479  F349  85 20              STA DRVST
9480  F34B              ;
9481  F34B  A0 05              LDY #NUMJOB-1
9482  F34D  84 3F              STY JOBN
9483  F34F              ;
9484  F34F  20 E1 F3    QUE10  JSR SETJB
9485  F352  30 1A              BMI QUE20
9486  F354              ;
9487  F354  C6 3F       QUE05  DEC JOBN
9488  F356  10 F7              BPL QUE10
9489  F358              ;
9490  F358              ;
9491  F358  A4 41              LDY NXTJOB
9492  F35A  20 E3 F3           JSR SETJB1
9493  F35D              ;
9494  F35D  A5 42              LDA NXTRK
9495  F35F  85 4A              STA STEPS
9496  F361  06 4A              ASL STEPS      ; STEPS*2
9497  F363              ;
9498  F363  A9 60              LDA #$60       ; SET STATUS = STEPPING
9499  F365  85 20              STA DRVST
9500  F367              ;
9501  F367              ;
9502  F367              ;
9503  F367  B1 32              LDA (HDRPNT),Y ; GET DEST TRACK #
9504  F369  85 22              STA DRVTRK
9505  F36B  4C EA F9    FIN    JMP END
9506  F36E              ;
9507  F36E              ;
9508  F36E  29 01       QUE20  AND #1          ; TEST IF SAME DRIVE
9509  F370  C5 3D              CMP DRIVE
9510  F372  D0 E0              BNE QUE05
9511  F374              ;
9512  F374  A5 22              LDA DRVTRK
9513  F376  F0 12              BEQ GOTU       ; UNINIT. TRACK #
```

```
LINE# LOC   CODE        LINE


9514  F378              ;
9515  F378  38                 SEC             ; CALC DISTANCE TO TRACK
9516  F379  F1 32              SBC (HDRPNT),Y
9517  F37B  F0 0D              BEQ GOTU        ; ON TRACK
9518  F37D              ;
9519  F37D  49 FF              EOR #$FF        ; 2'S COMP
9520  F37F  85 42              STA NXTRK
9521  F381  E6 42              INC NXTRK
9522  F383              ;
9523  F383  A5 3F              LDA JOBN        ; SAVE JOB# AND DIST TO TRCK
9524  F385  85 41              STA NXTJOB
9525  F387              ;
9526  F387  4C 54 F3           JMP QUE05
9527  F38A              ;
9528  F38A              ;
9529  F38A              ;
9530  F38A              ;
9531  F38A  A2 04       GOTU   LDX #4           ; SET TRACK AND SECTR
9532  F38C  B1 32              LDA (HDRPNT),Y
9533  F38E  85 40              STA TRACC
9534  F390  DD 25 FF    GOTU10 CMP TRKNUM-1,X
9535  F393  CA                 DEX
9536  F394  B0 FA              BCS GOTU10
9537  F396              ;
9538  F396  BD 20 FF           LDA NUMSEC,X
9539  F399  85 43              STA SECTR
9540  F39B              ;
9541  F39B  8A                 TXA             ; SET DENSITY
9542  F39C  0A                 ASL A
9543  F39D  0A                 ASL A
9544  F39E  0A                 ASL A
9545  F39F  0A                 ASL A
9546  F3A0  0A                 ASL A
9547  F3A1  85 44              STA WORK
9548  F3A3              ;
9549  F3A3  AD 00 1C           LDA DSKCNT
9550  F3A6  29 9F              AND #$9F        ; CLEAR DENSITY BITS
9551  F3A8  05 44              ORA WORK
9552  F3AA  8D 00 1C           STA DSKCNT
9553  F3AD              ;
9554  F3AD  A6 3D              LDX DRIVE       ; DRIVE NUM IN .X
9555  F3AF              ;
9556  F3AF  A5 45              LDA JOB         ; YES, GO DO THE JOB
9557  F3B1  C9 40              CMP #BUMPC      ; TEST FOR BUMP
9558  F3B3  F0 15              BEQ BMP
9559  F3B5              ;
9560  F3B5              ;
9561  F3B5  C9 60       EXE    CMP #EXECD       ; TEST IF EXECUTE
9562  F3B7  F0 03              BEQ EX
9563  F3B9              ;
9564  F3B9  4C FF F3           JMP SEAK        ; DO A SECTOR SEEK
9565  F3BC              ;
9566  F3BC  A5 3F       EX     LDA JOBN         ; JUMP TO BUFFER
9567  F3BE  18          EX2    CLC
9568  F3BF  69 03              ADC #>BUFS
```

```
LINE# LOC   CODE        LINE


9569  F3C1  85 31              STA BUFPNT+1
9570  F3C3  A9 00              LDA #<BUFS
9571  F3C5  85 30              STA BUFPNT
9572  F3C7  6C 30 00    EX3    JMP (BUFPNT)
9573  F3CA              ;
9574  F3CA              ;
9575  F3CA              BMP
9576  F3CA  A9 60              LDA #$60       ; SET STATUS=STEPPING
9577  F3CC  85 20              STA DRVST
9578  F3CE              ;
9579  F3CE  AD 00 1C           LDA DSKCNT
9580  F3D1  29 FC              AND #$FF-$03   ; SET PHASE A
9581  F3D3  8D 00 1C           STA DSKCNT
9582  F3D6              ;
9583  F3D6              ;
9584  F3D6              ;
9585  F3D6  A9 A4              LDA #256-92    ; STEP BACK 45 TRAKS
9586  F3D8  85 4A              STA STEPS
9587  F3DA              ;
9588  F3DA  A9 01              LDA #1         ; DRVTRK IS NOW 1
9589  F3DC  85 22              STA DRVTRK
9590  F3DE              ;
9591  F3DE  4C B7 F9           JMP ERRR       ; JOB DONE RETURN 1
9592  F3E1              ;
9593  F3E1              ;
9594  F3E1              ;
9595  F3E1              ;
9596  F3E1              ;
9597  F3E1  A4 3F       SETJB  LDY JOBN
9598  F3E3  B9 00 00    SETJB1 LDA JOBS,Y
9599  F3E6  48                 PHA
9600  F3E7  10 10              BPL SETJ10     ; NO JOB HERE
9601  F3E9              ;
9602  F3E9  29 78              AND #$78
9603  F3EB  85 45              STA JOB
9604  F3ED  98                 TYA
9605  F3EE  0A                 ASL A
9606  F3EF  69 06              ADC #<HDRS
9607  F3F1  85 32              STA HDRPNT
9608  F3F3  98                 TYA            ; POINT AT BUFFER
9609  F3F4  18                 CLC
9610  F3F5  69 03              ADC #>BUFS
9611  F3F7  85 31              STA BUFPNT+1
9612  F3F9              ;
9613  F3F9              ;
9614  F3F9  A0 00       SETJ10 LDY #0
9615  F3FB  84 30              STY BUFPNT
9616  F3FD              ;
9617  F3FD  68                 PLA
9618  F3FE  60                 RTS
9619  F3FF              ;
9620  F3FF              ;
9621  F3FF              ;
9622  F3FF              ;.END
9622  F3FF              ;
```

```
LINE# LOC   CODE        LINE

9623  F3FF                     .LIB LCCSEEK
```

```
LINE# LOC   CODE        LINE


9625  F3FF              ;
9626  F3FF              ;
9627  F3FF              ;
9628  F3FF  A2 5A       SEAK   LDX #90          ; SEARCH 90 HEADERS
9629  F401  86 4B              STX TMP
9630  F403              ;
9631  F403  A2 00              LDX #0          ;READ IN 8 GCR BYTES
9632  F405              ;
9633  F405  A9 52              LDA #$52        ; HEADER BLOCK ID
9634  F407  85 24              STA STAB
9635  F409              ;
9636  F409  20 A4 F5    SEEK10 JSR SYNC         ; FIND SYNC CHARACTER
9637  F40C              ;
9638  F40C  50 FE              BVC *           ; WAIT FOR BLOCK ID
9639  F40E  B8                 CLV
9640  F40F              ;
9641  F40F  AD 01 1C           LDA DATA2
9642  F412  C5 24              CMP STAB        ; TEST IF HEADER BLOCK
9643  F414  D0 3F              BNE SEEK20      ; NOT HEADER
9644  F416              ;
9645  F416  50 FE       SEEK15 BVC *
9646  F418  B8                 CLV             ; READ IN GCR HEADER
9647  F419              ;
9648  F419  AD 01 1C           LDA DATA2
9649  F41C  95 25              STA STAB+1,X
9650  F41E              ;
9651  F41E  E8                 INX
9652  F41F  E0 07              CPX #7
9653  F421  D0 F3              BNE SEEK15
9654  F423              ;
9655  F423  20 E5 F4           JSR CNVBIN      ; CONVERT HEADER IN STAB TO BINARY
                                                IN HEADER
9656  F426              ;
9657  F426  A0 04              LDY #4          ; COMPUTE CHECKSUM
9658  F428  A9 00              LDA #0
9659  F42A              ;
9660  F42A  59 16 00    SEEK30 EOR HEADER,Y
9661  F42D  88                 DEY
9662  F42E  10 FA              BPL SEEK30
9663  F430              ;
9664  F430  C9 00              CMP #0          ; TEST IF OK
9665  F432  D0 38              BNE CSERR       ; NOPE, CHECKSUM ERROR IN HEADER
9666  F434              ;
9667  F434  A6 3E              LDX CDRIVE      ; UPDATE DRIVE TRACK#
9668  F436  A5 18              LDA HEADER+2
9669  F438  95 22              STA DRVTRK,X
9670  F43A              ;
9671  F43A  A5 45              LDA JOB         ; TEST IF A SEEK JOB
9672  F43C  C9 30              CMP #$30
9673  F43E  F0 1E              BEQ ESEEK
9674  F440              ;
9675  F440  A5 3E              LDA CDRIVE
9676  F442  0A                 ASL A           ; TEST IF CORRECT ID
9677  F443  A8                 TAY
9678  F444  B9 12 00           LDA DSKID,Y
9679  F447  C5 16              CMP HEADER
```

```
LINE# LOC   CODE        LINE


9680  F449  D0 1E              BNE BADID
9681  F44B  B9 13 00           LDA DSKID+1,Y
9682  F44E  C5 17              CMP HEADER+1
9683  F450  D0 17              BNE BADID
9684  F452              ;
9685  F452  4C 71 F4           JMP WSECT      ; FIND BEST SECTOR TO SERVICE
9686  F455              ;
9687  F455              ;
9688  F455  C6 4B       SEEK20 DEC TMP         ; SEARCH MORE?
9689  F457  D0 B0              BNE SEEK10     ; YES
9690  F459              ;
9691  F459  A9 02              LDA #2         ; CANT FIND A SECTOR
9692  F45B  20 B7 F9           JSR ERRR
9693  F45E              ;
9694  F45E              ;
9695  F45E  A5 16       ESEEK  LDA HEADER      ; HARRIS FIX....
9696  F460  85 12              STA DSKID      ; ....
9697  F462  A5 17              LDA HEADER+1   ; ....
9698  F464  85 13              STA DSKID+1    ; ....
9699  F466              ;
9700  F466  A9 01       DONE   LDA #1          ; RETURN OK CODE
9701  F468  2C                 .BYTE $2C
9702  F469              ;
9703  F469  A9 0B       BADID  LDA #11         ; DISK ID MISMATCH
9704  F46B  2C                 .BYTE $2C
9705  F46C              ;
9706  F46C  A9 09       CSERR  LDA #9          ; CHECKSUM ERROR IN HEADER
9707  F46E  4C B7 F9           JMP ERRR
9708  F471              ;
9709  F471              ;
9710  F471              ;
9711  F471  A9 7F       WSECT  LDA #$7F        ; FIND BEST JOB
9712  F473  85 4C              STA CSECT
9713  F475              ;
9714  F475  A5 19              LDA HEADER+3   ; GET UPCOMING SECTOR #
9715  F477  18                 CLC
9716  F478  69 02              ADC #2
9717  F47A  C5 43              CMP SECTR
9718  F47C  90 02              BCC L460
9719  F47E              ;
9720  F47E  E5 43              SBC SECTR      ; WRAP AROUND
9721  F480              ;
9722  F480  85 4D       L460   STA NEXTS       ; NEXT SECTOR
9723  F482              ;
9724  F482  A2 05              LDX #NUMJOB-1
9725  F484  86 3F              STX JOBN
9726  F486              ;
9727  F486  A2 FF              LDX #$FF
9728  F488              ;
9729  F488  20 E1 F3    L480   JSR SETJB
9730  F48B  10 44              BPL L470
9731  F48D              ;
9732  F48D  85 44              STA WORK
9733  F48F  29 01              AND #1
9734  F491  C5 3E              CMP CDRIVE     ; TEST IF SAME DRIVE
```

```
LINE# LOC   CODE        LINE


9735  F493  D0 3C              BNE L470       ; NOPE
9736  F495              ;
9737  F495  A0 00              LDY #0         ; TEST IF SAME TRACK
9738  F497  B1 32              LDA (HDRPNT),Y
9739  F499  C5 40              CMP TRACC
9740  F49B  D0 34              BNE L470
9741  F49D              ;
9742  F49D  A5 45              LDA JOB        ; TEST IF EXECUTE JOB
9743  F49F  C9 60              CMP #EXECD
9744  F4A1  F0 0C              BEQ L465
9745  F4A3              ;
9746  F4A3  A0 01              LDY #1
9747  F4A5  38                 SEC
9748  F4A6  B1 32              LDA (HDRPNT),Y
9749  F4A8  E5 4D              SBC NEXTS
9750  F4AA  10 03              BPL L465
9751  F4AC              ;
9752  F4AC  18                 CLC
9753  F4AD  65 43              ADC SECTR
9754  F4AF              ;
9755  F4AF  C5 4C       L465   CMP CSECT
9756  F4B1  B0 1E              BCS L470
9757  F4B3              ;
9758  F4B3  48                 PHA            ; SAVE IT
9759  F4B4  A5 45              LDA JOB
9760  F4B6  F0 14              BEQ TSTRDJ     ; MUST BE A READ
9761  F4B8              ;
9762  F4B8  68                 PLA
9763  F4B9  C9 09              CMP #WRTMIN    ; .IF(CSECT<9)RETURN;
9764  F4BB  90 14              BCC L470
9765  F4BD              ;
9766  F4BD  C9 0C              CMP #WRTMAX    ; .IF(CSECT>12)RETURN;
9767  F4BF  B0 10              BCS L470
9768  F4C1              ;
9769  F4C1  85 4C       DOITT  STA CSECT       ; ITS BETTER
9770  F4C3  A5 3F              LDA JOBN
9771  F4C5  AA                 TAX
9772  F4C6  69 03              ADC #>BUFS
9773  F4C8  85 31              STA BUFPNT+1
9774  F4CA              ;
9775  F4CA  D0 05              BNE L470
9776  F4CC              ;
9777  F4CC  68          TSTRDJ PLA
9778  F4CD  C9 06              CMP #RDMAX     ; IF(CSECT>6)RETURN;
9779  F4CF  90 F0              BCC DOITT
9780  F4D1              ;
9781  F4D1              ;
9782  F4D1  C6 3F       L470   DEC JOBN
9783  F4D3  10 B3              BPL L480
9784  F4D5              ;
9785  F4D5  8A                 TXA            ; TEST IF A JOB TO DO
9786  F4D6  10 03              BPL L490
9787  F4D8              ;
9788  F4D8  4C EA F9           JMP END        ; NO JOB FOUND
9789  F4DB              ;
```

```
LINE# LOC   CODE         LINE


9790  F4DB  86 3F        L490   STX JOBN
9791  F4DD  20 E1 F3            JSR SETJB
9792  F4E0  A5 45               LDA JOB
9793  F4E2  4C 18 F5            JMP REED
9794  F4E5               ;
9795  F4E5               ;
9796  F4E5               ;
9797  F4E5               ;
9798  F4E5  A5 30        CNVBIN LDA BUFPNT
9799  F4E7  48                  PHA
9800  F4E8  A5 31               LDA BUFPNT+1
9801  F4EA  48                  PHA             ; SAVE BUFFER POINTER
9802  F4EB               ;
9803  F4EB  A9 24               LDA #<STAB
9804  F4ED  85 30               STA BUFPNT      ; POINT AT GCR CODE
9805  F4EF  A9 00               LDA #>STAB
9806  F4F1  85 31               STA BUFPNT+1
9807  F4F3               ;
9808  F4F3  A9 00               LDA #0
9809  F4F5  85 34               STA GCRPNT
9810  F4F7               ;
9811  F4F7  20 34 F8            JSR GET4GB      ; CONVERT 4 BYTES
9812  F4FA               ;
9813  F4FA  A5 55               LDA BTAB+3
9814  F4FC  85 18               STA HEADER+2
9815  F4FE               ;
9816  F4FE  A5 54               LDA BTAB+2
9817  F500  85 19               STA HEADER+3
9818  F502               ;
9819  F502  A5 53               LDA BTAB+1
9820  F504  85 1A               STA HEADER+4
9821  F506               ;
9822  F506               ;
9823  F506  20 34 F8            JSR GET4GB      ; GET 2 MORE
9824  F509               ;
9825  F509  A5 52               LDA BTAB        ; GET ID
9826  F50B  85 17               STA HEADER+1
9827  F50D  A5 53               LDA BTAB+1
9828  F50F  85 16               STA HEADER
9829  F511               ;
9830  F511  68                  PLA
9831  F512  85 31               STA BUFPNT+1    ; RESTORE POINTER
9832  F514  68                  PLA
9833  F515  85 30               STA BUFPNT
9834  F517               ;
9835  F517  60                  RTS
9836  F518               ;
9837  F518               ;
9838  F518               ;
9839  F518               ;
9840  F518               ;
9841  F518               ;
9842  F518               ;
9843  F518               ;.END
9843  F518               ;
```

```
LINE# LOC   CODE        LINE

9844  F518                     .LIB LCCREAD
```

```
LINE# LOC   CODE         LINE


9846  F518               ;
9847  F518               ;
9848  F518               ;
9849  F518               ;
9850  F518               ;
9851  F518               ; *READ
9852  F518               ;
9853  F518               ; READ IN TRACK,SECTOR SPECIFIED
9854  F518               ; IN HEADER
9855  F518               ;
9856  F518               ;
9857  F518  C9 00        REED   CMP #0          ; TEST IF READ JOB
9858  F51A  F0 03               BEQ READ01     ; GO TEST IF WRITE
9859  F51C  4C BC F5            JMP WRIGHT
9860  F51F               ;
9861  F51F  20 58 F5     READ01 JSR DSTRT       ; FIND HEADER AND START READING
                                                DATA
9862  F522               ;
9863  F522  50 FE        READ11 BVC *           ; WAIT FOR BYTE
9864  F524  B8                  CLV
9865  F525               ;
9866  F525  AD 01 1C            LDA DATA2      ; STORE AWAY DATA
9867  F528  91 30               STA (BUFPNT),Y ; IN DATA BUFFER
9868  F52A  C8                  INY
9869  F52B  D0 F5               BNE READ11
9870  F52D               ;
9871  F52D  A0 BA               LDY #255-TOPRD ; STORE REST IN OVERFLOW BUFFER
9872  F52F               ;
9873  F52F  50 FE        READ20 BVC *
9874  F531  B8                  CLV
9875  F532               ;
9876  F532  AD 01 1C            LDA DATA2
9877  F535  99 00 01            STA OVRBUF,Y
9878  F538  C8                  INY
9879  F539  D0 F4               BNE READ20
9880  F53B               ;
9881  F53B  20 2E F9            JSR GCRBIN     ; CONVERT BUFFER TO BINARY
9882  F53E               ;
9883  F53E  A5 38               LDA BID        ; TEST IF ITS A DATA BLOCK
9884  F540  C5 47               CMP DBID
9885  F542  F0 05               BEQ READ28
9886  F544               ;
9887  F544  A9 04               LDA #4         ; NOT A DATA BLOCK, GET ERROR#
9888  F546  4C B7 F9            JMP ERRR
9889  F549               ;
9890  F549  20 37 F6     READ28 JSR CHKBLK      ; CALC CHECKSUM
9891  F54C               ;
9892  F54C  C5 3A               CMP CHKSUM
9893  F54E  F0 03               BEQ READ40
9894  F550               ;
9895  F550  A9 05               LDA #5         ; DATA BLOCK CHECKSUM ERROR
9896  F552  2C                  .BYTE $2C      ; SKIP NEXT INSTRUCTION
9897  F553               ;
9898  F553  A9 01        READ40 LDA #1          ; READ DATA BLOCK OK
9899  F555  4C B7 F9            JMP ERRR
9900  F558               ;
```

```
LINE# LOC   CODE        LINE


9901  F558              ;
9902  F558              ;
9903  F558  20 5E F5    DSTRT  JSR SRCH        ; FIND HEADER
9904  F55B  4C A4 F5           JMP SYNC       ; AND THEN DATA BLOCK SYNC
9905  F55E              ;
9906  F55E              ;
9907  F55E  A5 3D       SRCH   LDA DRIVE       ; CREATE HEADER IMAGE
9908  F560  0A                 ASL A
9909  F561  AA                 TAX
9910  F562              ;
9911  F562  B5 12              LDA DSKID,X    ; GET MASTER ID FOR THE DRIVE
9912  F564  85 16              STA HEADER
9913  F566  B5 13              LDA DSKID+1,X
9914  F568  85 17              STA HEADER+1
9915  F56A              ;
9916  F56A  A0 00              LDY #0         ; GET TRACK,SECTOR
9917  F56C  B1 32              LDA (HDRPNT),Y
9918  F56E  85 18              STA HEADER+2
9919  F570  C8                 INY
9920  F571  B1 32              LDA (HDRPNT),Y
9921  F573  85 19              STA HEADER+3
9922  F575              ;
9923  F575  A9 00              LDA #0
9924  F577                                    ; CREATE HEADER CHECKSUM
9925  F577  45 16              EOR HEADER
9926  F579  45 17              EOR HEADER+1
9927  F57B  45 18              EOR HEADER+2
9928  F57D  45 19              EOR HEADER+3
9929  F57F              ;
9930  F57F  85 1A              STA HEADER+4   ; STORE THE CHECKSUM
9931  F581              ;
9932  F581  20 82 F9           JSR CONHDR     ; CONVERT HEADER TO GCR
9933  F584              ;
9934  F584  A2 5A              LDX #90        ; SEARCH 90 SYNC CHARS
9935  F586              ;
9936  F586  20 A4 F5    SRCH20 JSR SYNC        ; FIND SYNC
9937  F589              ;
9938  F589  A0 00              LDY #0         ; TEST 8 GCR BYTES
9939  F58B              ;
9940  F58B  50 FE       SRCH25 BVC *
9941  F58D  B8                 CLV            ; WAIT FOR BYTE
9942  F58E              ;
9943  F58E  AD 01 1C           LDA DATA2
9944  F591  D9 24 00           CMP STAB,Y     ; TEST IF THE SAME
9945  F594  D0 06              BNE SRCH30     ; NOPE
9946  F596              ;
9947  F596  C8                 INY
9948  F597  C0 08              CPY #8
9949  F599  D0 F0              BNE SRCH25
9950  F59B              ;
9951  F59B  60                 RTS
9952  F59C              ;
9953  F59C              ;
9954  F59C  CA          SRCH30 DEX             ; TRY AGAIN
9955  F59D  D0 E7              BNE SRCH20
```

```
LINE# LOC   CODE         LINE


9956  F59F               ;
9957  F59F  A9 02               LDA #2          ; CANT FIND THIS HEADER
9958  F5A1  4C B7 F9     ERR    JMP ERRR
9959  F5A4               ;
9960  F5A4               ;
9961  F5A4               ;
9962  F5A4               ;
9963  F5A4               SYNC
9964  F5A4               ;
9965  F5A4  A9 D0               LDA #$80+80     ; WAIT 20 MS FOR SYNC MAX
9966  F5A6  8D 05 18            STA TIMER1
9967  F5A9               ;
9968  F5A9  A9 03               LDA #3          ; ERROR CODE FOR NO SYNC
9969  F5AB               ;
9970  F5AB  2C 05 18     SYNC10 BIT TIMER1       ; TEST FOR TIME OUT
9971  F5AE  10 F1               BPL ERR
9972  F5B0  2C 00 1C            BIT DSKCNT      ; TEST FOR SYNC
9973  F5B3  30 F6               BMI SYNC10
9974  F5B5               ;
9975  F5B5               ;
9976  F5B5  AD 01 1C            LDA DATA2       ; RESET PA LATCH
9977  F5B8  B8                  CLV
9978  F5B9  A0 00               LDY #0          ; CLEAR POINTER
9979  F5BB  60                  RTS
9980  F5BC               ;
9981  F5BC               ;
9982  F5BC               ;
9983  F5BC               ;.END
9983  F5BC               ;
9984  F5BC                      .LIB LCCWRT
```

```
LINE# LOC   CODE        LINE


9986  F5BC              ;
9987  F5BC              ;
9988  F5BC              ;
9989  F5BC              ; * WRITE JOB
9990  F5BC              ;
9991  F5BC              ; WRITE OUT DATA BUFFER
9992  F5BC              ;
9993  F5BC              ;
9994  F5BC  C9 10       WRIGHT CMP #$10        ; TEST IF WRITE
9995  F5BE  F0 03              BEQ WRT05
9996  F5C0              ;
9997  F5C0  4C DF F6           JMP VRFY
9998  F5C3              ;
9999  F5C3  20 37 F6    WRT05  JSR CHKBLK      ; GET BLOCK CHECKSUM
0000  F5C6  85 3A              STA CHKSUM
0001  F5C8              ;
0002  F5C8  AD 00 1C           LDA DSKCNT     ; TEST FOR WRITE PROTECT
0003  F5CB  29 10              AND #$10
0004  F5CD  D0 05              BNE WRT10      ; NOT PROTECTED
0005  F5CF              ;
0006  F5CF  A9 08              LDA #8         ; WRITE PROTECT ERROR
0007  F5D1  4C B7 F9           JMP ERRR
0008  F5D4              ;
0009  F5D4  20 DD F7    WRT10  JSR BINGCR      ; CONVERT BUFFER TO WRITE IMAGE
0010  F5D7              ;
0011  F5D7  20 5E F5           JSR SRCH       ; FIND HEADER
0012  F5DA              ;
0013  F5DA  A2 08              LDX #GAP1-2    ; WAIT OUT HEADER GAP (1541 HAS 9)
0014  F5DC              ;
0015  F5DC  50 FE       WRT20  BVC *
0016  F5DE  B8                 CLV
0017  F5DF              ;
0018  F5DF  CA                 DEX            ; TEST IF DONE YET
0019  F5E0  D0 FA              BNE WRT20
0020  F5E2              ;
0021  F5E2  A9 FF              LDA #$FF
0022  F5E4  8D 03 1C           STA DDRA2
0023  F5E7              ;
0024  F5E7  AD 0C 1C           LDA PCR2       ; SET WRITE MODE
0025  F5EA  29 1F              AND #$FF-$E0
0026  F5EC  09 C0              ORA #$C0
0027  F5EE  8D 0C 1C           STA PCR2
0028  F5F1              ;
0029  F5F1  A9 FF              LDA #$FF       ; WRITE 5 GCR SYNC
0030  F5F3  A2 05              LDX #NUMSYN    ;
0031  F5F5  8D 01 1C           STA DATA2
0032  F5F8  B8                 CLV
0033  F5F9              ;
0034  F5F9  50 FE       WRTSNC BVC *
0035  F5FB              ;
0036  F5FB  B8                 CLV
0037  F5FC  CA                 DEX
0038  F5FD  D0 FA              BNE WRTSNC
0039  F5FF              ;
0040  F5FF  A0 BB              LDY #256-TOPWRT ; WRITE OUT OVERFLOW BUFFER
```

```
LINE# LOC   CODE        LINE


0041  F601              ;
0042  F601  B9 00 01    WRT30  LDA OVRBUF,Y    ; GET A CHAR
0043  F604  50 FE              BVC *          ; WAIT UNTIL READY
0044  F606  B8                 CLV
0045  F607              ;
0046  F607  8D 01 1C           STA DATA2      ; STUFF IT
0047  F60A  C8                 INY
0048  F60B  D0 F4              BNE WRT30      ; DO NEXT CHAR
0049  F60D              ;
0050  F60D              ; WRITE REST OF BUFFER
0051  F60D              ;
0052  F60D  B1 30       WRT40  LDA (BUFPNT),Y  ; NOW DO BUFFER
0053  F60F  50 FE              BVC *          ; WAIT UNTIL READY
0054  F611  B8                 CLV
0055  F612              ;
0056  F612  8D 01 1C           STA DATA2      ; STUFF IT AGAIN
0057  F615  C8                 INY
0058  F616              ; TEST IF DONE
0059  F616  D0 F5              BNE WRT40      ; DO THE WHOLE THING
0060  F618              ;
0061  F618  50 FE              BVC *          ; WAIT FOR LAST CHAR TO WRITE OUT
0062  F61A              ;
0063  F61A              ;
0064  F61A  AD 0C 1C           LDA PCR2       ; GOTO READ MODE
0065  F61D  09 E0              ORA #$E0
0066  F61F  8D 0C 1C           STA PCR2
0067  F622              ;
0068  F622  A9 00              LDA #0         ; MAKE DATA2 INPUT $00
0069  F624  8D 03 1C           STA DDRA2
0070  F627              ;
0071  F627  20 40 F6           JSR WTOBIN     ; CONVERT WRITE IMAGE TO BINARY
0072  F62A              ;
0073  F62A  A4 3F              LDY JOBN       ; MAKE JOB A VERIFY
0074  F62C  B9 00 00           LDA JOBS,Y
0075  F62F  49 30              EOR #$30
0076  F631  99 00 00           STA JOBS,Y
0077  F634              ;
0078  F634  4C FF F3           JMP SEAK       ; SCAN JOB QUEUE
0079  F637              ;
0080  F637              ;
0081  F637  A9 00       CHKBLK LDA #0          ; GET EOR CHECKSUM
0082  F639  A8                 TAY
0083  F63A              ;
0084  F63A  51 30       CHKB10 EOR (BUFPNT),Y
0085  F63C  C8                 INY
0086  F63D  D0 FB              BNE CHKB10
0087  F63F              ;
0088  F63F  60                 RTS            ; RETURN CHECKSUM IN .A
0089  F640              ;
0090  F640              ;
0091  F640              ;
0092  F640              ;
0093  F640              ; * WTOBIN
0094  F640              ;
0095  F640              ; CONVERT WRITE IMAGE BACK TO
```

```
LINE# LOC   CODE        LINE


0096  F640              ; BINARY DATA
0097  F640              ;
0098  F640              ;
0099  F640  A9 00       WTOBIN LDA #0           ; INIT POINTER
0100  F642  85 2E              STA SAVPNT
0101  F644  85 30              STA BUFPNT     ; LSB
0102  F646  85 4F              STA NXTPNT
0103  F648              ;
0104  F648  A5 31              LDA BUFPNT+1   ; GOTO BUFFER NEXT
0105  F64A  85 4E              STA NXTBF
0106  F64C              ;
0107  F64C  A9 01              LDA #>OVRBUF   ; USE OVERFLOW FIRST
0108  F64E  85 31              STA BUFPNT+1
0109  F650  85 2F              STA SAVPNT+1
0110  F652              ;
0111  F652  A9 BB              LDA #256-TOPWRT
0112  F654  85 34              STA GCRPNT     ; GET HERE FIRST
0113  F656  85 36              STA BYTCNT     ; STORE HERE
0114  F658              ;
0115  F658  20 34 F8           JSR GET4GB     ; GET FIRST FOUR- ID AND 3 DATA
0116  F65B              ;
0117  F65B  A5 52              LDA BTAB       ; SAVE BID
0118  F65D  85 38              STA BID
0119  F65F              ;
0120  F65F  A4 36              LDY BYTCNT
0121  F661              ;
0122  F661  A5 53              LDA BTAB+1
0123  F663  91 2E              STA (SAVPNT),Y
0124  F665  C8                 INY
0125  F666              ;
0126  F666  A5 54              LDA BTAB+2
0127  F668  91 2E              STA (SAVPNT),Y
0128  F66A  C8                 INY
0129  F66B              ;
0130  F66B  A5 55              LDA BTAB+3
0131  F66D  91 2E              STA (SAVPNT),Y
0132  F66F  C8                 INY
0133  F670              ;
0134  F670  84 36              STY BYTCNT
0135  F672              ;
0136  F672  20 34 F8    WTOB14 JSR GET4GB       ; DO REST OF OVERFLOW BUFFER
0137  F675              ;
0138  F675  A4 36              LDY BYTCNT
0139  F677              ;
0140  F677  A5 52              LDA BTAB
0141  F679  91 2E              STA (SAVPNT),Y
0142  F67B  C8                 INY
0143  F67C              ;
0144  F67C  A5 53              LDA BTAB+1
0145  F67E  91 2E              STA (SAVPNT),Y
0146  F680  C8                 INY
0147  F681  F0 0E              BEQ WTOB50
0148  F683              ;
0149  F683  A5 54              LDA BTAB+2
0150  F685  91 2E              STA (SAVPNT),Y
```

```
LINE# LOC   CODE        LINE


0151  F687  C8                 INY
0152  F688              ;
0153  F688  A5 55              LDA BTAB+3
0154  F68A  91 2E              STA (SAVPNT),Y
0155  F68C  C8                 INY
0156  F68D              ;
0157  F68D  84 36              STY BYTCNT
0158  F68F  D0 E1              BNE WTOB14     ; JMP
0159  F691              ;
0160  F691              WTOB50
0161  F691              ;
0162  F691  A5 54              LDA BTAB+2
0163  F693  91 30              STA (BUFPNT),Y
0164  F695  C8                 INY
0165  F696              ;
0166  F696  A5 55              LDA BTAB+3
0167  F698  91 30              STA (BUFPNT),Y
0168  F69A  C8                 INY
0169  F69B              ;
0170  F69B  84 36              STY BYTCNT
0171  F69D              ;
0172  F69D  20 34 F8    WTOB53 JSR GET4GB
0173  F6A0              ;
0174  F6A0  A4 36              LDY BYTCNT
0175  F6A2              ;
0176  F6A2  A5 52              LDA BTAB
0177  F6A4  91 30              STA (BUFPNT),Y
0178  F6A6  C8                 INY
0179  F6A7              ;
0180  F6A7  A5 53              LDA BTAB+1
0181  F6A9  91 30              STA (BUFPNT),Y
0182  F6AB  C8                 INY
0183  F6AC              ;
0184  F6AC  A5 54              LDA BTAB+2
0185  F6AE  91 30              STA (BUFPNT),Y
0186  F6B0  C8                 INY
0187  F6B1              ;
0188  F6B1  A5 55              LDA BTAB+3
0189  F6B3  91 30              STA (BUFPNT),Y
0190  F6B5  C8                 INY
0191  F6B6              ;
0192  F6B6  84 36              STY BYTCNT
0193  F6B8  C0 BB              CPY #187
0194  F6BA  90 E1              BCC WTOB53
0195  F6BC              ;
0196  F6BC  A9 45       WTOB52 LDA #69          ; MOVE BUFFER UP
0197  F6BE  85 2E              STA SAVPNT
0198  F6C0              ;
0199  F6C0  A5 31              LDA BUFPNT+1
0200  F6C2  85 2F              STA SAVPNT+1
0201  F6C4              ;
0202  F6C4  A0 BA              LDY #256-TOPWRT-1
0203  F6C6              ;
0204  F6C6  B1 30       WTOB55 LDA (BUFPNT),Y
0205  F6C8  91 2E              STA (SAVPNT),Y
```

```
LINE# LOC   CODE        LINE


0206  F6CA              ;
0207  F6CA  88                 DEY
0208  F6CB  D0 F9              BNE WTOB55
0209  F6CD              ;
0210  F6CD  B1 30              LDA (BUFPNT),Y
0211  F6CF  91 2E              STA (SAVPNT),Y
0212  F6D1              ;
0213  F6D1  A2 BB              LDX #256-TOPWRT ; MOVE OVERFLOW OVER TO BUFFER
0214  F6D3              ;
0215  F6D3  BD 00 01    WTOB57 LDA OVRBUF,X
0216  F6D6  91 30              STA (BUFPNT),Y
0217  F6D8              ;
0218  F6D8  C8                 INY
0219  F6D9  E8                 INX
0220  F6DA  D0 F7              BNE WTOB57
0221  F6DC              ;
0222  F6DC  86 50              STX GCRFLG     ; CLEAR BUFFER GCR FLAG
0223  F6DE              ;
0224  F6DE              ;
0225  F6DE  60                 RTS
0226  F6DF              ;
0227  F6DF              ;
0228  F6DF              ;
0229  F6DF              ;
0230  F6DF              ;
0231  F6DF              ; * VERIFY DATA BLOCK
0232  F6DF              ;
0233  F6DF              ; CONVERT TO GCR VERIFY IMAGE
0234  F6DF              ;
0235  F6DF              ; TEST AGAINST DATA BLOCK
0236  F6DF              ;
0237  F6DF              ; CONVERT BACK TO BINARY
0238  F6DF              ;
0239  F6DF              ;
0240  F6DF  C9 20       VRFY   CMP #$20        ; TEST IF VERIFY
0241  F6E1  F0 03              BEQ VRF10
0242  F6E3              ;
0243  F6E3  4C 18 F7           JMP SECTSK
0244  F6E6              ;
0245  F6E6              VRF10
0246  F6E6              ;
0247  F6E6  20 37 F6           JSR CHKBLK     ; GET BLOCK CHECKSUM
0248  F6E9  85 3A              STA CHKSUM
0249  F6EB              ;
0250  F6EB  20 DD F7           JSR BINGCR     ; CONVERT TO VERIFY IMAGE
0251  F6EE              ;
0252  F6EE  20 58 F5           JSR DSTRT
0253  F6F1              ;
0254  F6F1  A0 BB              LDY #256-TOPWRT
0255  F6F3  B9 00 01    VRF15  LDA OVRBUF,Y    ; GET CHAR
0256  F6F6  50 FE              BVC *
0257  F6F8  B8                 CLV
0258  F6F9              ;
0259  F6F9  4D 01 1C           EOR DATA2      ; TEST IF SAME
0260  F6FC  D0 15              BNE VRF20      ; VERIFY ERROR
```

```
LINE# LOC   CODE        LINE


0261  F6FE              ;
0262  F6FE  C8                 INY
0263  F6FF  D0 F2              BNE VRF15      ; NEXT BYTE
0264  F701              ;
0265  F701              ;
0266  F701  B1 30       VRF30  LDA (BUFPNT),Y  ; NOW DO BUFFER
0267  F703              ;
0268  F703  50 FE              BVC *
0269  F705  B8                 CLV            ; WAIT FOR CHAR
0270  F706              ;
0271  F706  4D 01 1C           EOR DATA2      ; TEST IF SAME
0272  F709  D0 08              BNE VRF20      ; ERROR
0273  F70B              ;
0274  F70B  C8                 INY
0275  F70C  C0 FD              CPY #$FD       ; DONT TEST OFF BYTES
0276  F70E  D0 F1              BNE VRF30
0277  F710              ;
0278  F710              ;
0279  F710  4C 66 F4           JMP DONE       ; VERIFY OK
0280  F713              ;
0281  F713  A9 07       VRF20  LDA #7          ; VERIFY ERROR
0282  F715  4C B7 F9           JMP ERRR
0283  F718              ;
0284  F718              ;
0285  F718  20 5E F5    SECTSK JSR SRCH        ; SECTOR SEEK
0286  F71B  4C 66 F4           JMP DONE
0287  F71E              ;
0288  F71E              ;
0289  F71E              ;.END
0289  F71E              ;
0290  F71E                     .LIB LCCBINGCR
```

```
LINE# LOC   CODE        LINE


0292  F71E              ; FAST BINARY TO GCR
0293  F71E              ;
0294  F71E              ;
0295  F71E  A9 00       PUT4BG LDA #0          ; CLEAR TABLE
0296  F720  85 57              STA GTAB+1
0297  F722  85 5A              STA GTAB+4
0298  F724              ;
0299  F724  A4 34              LDY GCRPNT
0300  F726              ;
0301  F726  A5 52              LDA BTAB
0302  F728  29 F0              AND #$F0
0303  F72A  4A                 LSR A
0304  F72B  4A                 LSR A
0305  F72C  4A                 LSR A
0306  F72D  4A                 LSR A
0307  F72E  AA                 TAX
0308  F72F  BD CD F7           LDA BGTAB,X
0309  F732              ;
0310  F732  0A                 ASL A
0311  F733  0A                 ASL A
0312  F734  0A                 ASL A
0313  F735  85 56              STA GTAB
0314  F737              ;
0315  F737  A5 52              LDA BTAB
0316  F739  29 0F              AND #$0F
0317  F73B  AA                 TAX
0318  F73C  BD CD F7           LDA BGTAB,X
0319  F73F              ;
0320  F73F  6A                 ROR A
0321  F740  66 57              ROR GTAB+1
0322  F742  6A                 ROR A
0323  F743  66 57              ROR GTAB+1
0324  F745              ;
0325  F745  29 07              AND #$07
0326  F747  05 56              ORA GTAB
0327  F749  91 30              STA (BUFPNT),Y
0328  F74B              ;
0329  F74B  C8                 INY
0330  F74C              ;
0331  F74C  A5 53              LDA BTAB+1
0332  F74E  29 F0              AND #$F0
0333  F750  4A                 LSR A
0334  F751  4A                 LSR A
0335  F752  4A                 LSR A
0336  F753  4A                 LSR A
0337  F754  AA                 TAX
0338  F755  BD CD F7           LDA BGTAB,X
0339  F758              ;
0340  F758  0A                 ASL A
0341  F759  05 57              ORA GTAB+1
0342  F75B  85 57              STA GTAB+1
0343  F75D              ;
0344  F75D              ;
0345  F75D  A5 53              LDA BTAB+1
0346  F75F  29 0F              AND #$0F
```

```
LINE# LOC   CODE        LINE


0347  F761  AA                 TAX
0348  F762  BD CD F7           LDA BGTAB,X
0349  F765              ;
0350  F765  2A                 ROL A
0351  F766  2A                 ROL A
0352  F767  2A                 ROL A
0353  F768  2A                 ROL A
0354  F769  85 58              STA GTAB+2
0355  F76B              ;
0356  F76B  2A                 ROL A
0357  F76C  29 01              AND #1
0358  F76E  05 57              ORA GTAB+1
0359  F770  91 30              STA (BUFPNT),Y
0360  F772              ;
0361  F772  C8                 INY
0362  F773              ;
0363  F773  A5 54              LDA BTAB+2
0364  F775  29 F0              AND #$F0
0365  F777  4A                 LSR A
0366  F778  4A                 LSR A
0367  F779  4A                 LSR A
0368  F77A  4A                 LSR A
0369  F77B  AA                 TAX
0370  F77C  BD CD F7           LDA BGTAB,X
0371  F77F              ;
0372  F77F  18                 CLC
0373  F780  6A                 ROR A
0374  F781  05 58              ORA GTAB+2
0375  F783  91 30              STA (BUFPNT),Y
0376  F785  C8                 INY
0377  F786              ;
0378  F786  6A                 ROR A
0379  F787  29 80              AND #$80
0380  F789  85 59              STA GTAB+3
0381  F78B              ;
0382  F78B  A5 54              LDA BTAB+2
0383  F78D  29 0F              AND #$0F
0384  F78F  AA                 TAX
0385  F790  BD CD F7           LDA BGTAB,X
0386  F793  0A                 ASL A
0387  F794  0A                 ASL A
0388  F795  29 7C              AND #$7C
0389  F797  05 59              ORA GTAB+3
0390  F799  85 59              STA GTAB+3
0391  F79B              ;
0392  F79B  A5 55              LDA BTAB+3
0393  F79D  29 F0              AND #$F0
0394  F79F  4A                 LSR A
0395  F7A0  4A                 LSR A
0396  F7A1  4A                 LSR A
0397  F7A2  4A                 LSR A
0398  F7A3  AA                 TAX
0399  F7A4  BD CD F7           LDA BGTAB,X
0400  F7A7              ;
0401  F7A7  6A                 ROR A
```

```
LINE# LOC   CODE        LINE


0402  F7A8  66 5A              ROR GTAB+4
0403  F7AA  6A                 ROR A
0404  F7AB  66 5A              ROR GTAB+4
0405  F7AD  6A                 ROR A
0406  F7AE  66 5A              ROR GTAB+4
0407  F7B0              ;
0408  F7B0  29 03              AND #$03
0409  F7B2  05 59              ORA GTAB+3
0410  F7B4  91 30              STA (BUFPNT),Y
0411  F7B6  C8                 INY
0412  F7B7  D0 04              BNE BING35
0413  F7B9              ;
0414  F7B9  A5 2F              LDA SAVPNT+1
0415  F7BB  85 31              STA BUFPNT+1
0416  F7BD              ;
0417  F7BD              ;
0418  F7BD  A5 55       BING35 LDA BTAB+3
0419  F7BF  29 0F              AND #$0F
0420  F7C1  AA                 TAX
0421  F7C2  BD CD F7           LDA BGTAB,X
0422  F7C5  05 5A              ORA GTAB+4
0423  F7C7  91 30              STA (BUFPNT),Y
0424  F7C9  C8                 INY
0425  F7CA  84 34              STY GCRPNT
0426  F7CC  60                 RTS
0427  F7CD              ;
0428  F7CD              ;
0429  F7CD              ;
0430  F7CD              ; TABLE FOR BINARY TO GCR CONVERSION
0431  F7CD  0A          BGTAB  .BYTE $0A
0432  F7CE  0B                 .BYTE $0B
0433  F7CF  12                 .BYTE $12
0434  F7D0  13                 .BYTE $13
0435  F7D1  0E                 .BYTE $0E
0436  F7D2  0F                 .BYTE $0F
0437  F7D3  16                 .BYTE $16
0438  F7D4  17                 .BYTE $17
0439  F7D5  09                 .BYTE $09
0440  F7D6  19                 .BYTE $19
0441  F7D7  1A                 .BYTE $1A
0442  F7D8  1B                 .BYTE $1B
0443  F7D9  0D                 .BYTE $0D
0444  F7DA  1D                 .BYTE $1D
0445  F7DB  1E                 .BYTE $1E
0446  F7DC  15                 .BYTE $15
0447  F7DD              ;
0448  F7DD              ;
0449  F7DD              ;
0450  F7DD              ;******************************
0451  F7DD              ;*
0452  F7DD              ;*
0453  F7DD              ;* BINARY TO GCR CONVERSION
0454  F7DD              ;*
0455  F7DD              ;*
0456  F7DD              ;* DOES INPLACE CONVERSION OF
```

```
LINE# LOC   CODE        LINE


0457  F7DD              ;* BUFFER TO GCR USING OVERFLOW
0458  F7DD              ;* BLOCK
0459  F7DD              ;*
0460  F7DD              ;*
0461  F7DD              ;* CREATES WRITE IMAGE
0462  F7DD              ;*
0463  F7DD              ;* 1 BLOCK ID CHAR
0464  F7DD              ;* 256 DATA BYTES
0465  F7DD              ;* 1 CHECK SUM
0466  F7DD              ;* 2 OFF BYTES
0467  F7DD              ;* ---
0468  F7DD              ;* 260 BINARY BYTES
0469  F7DD              ;*
0470  F7DD              ;* 260 BINARY BYTES >> 325 GCR
0471  F7DD              ;*
0472  F7DD              ;* 325 = 256 + 69 OVERFLOW
0473  F7DD              ;*
0474  F7DD              ;*
0475  F7DD              ;********************************
0476  F7DD              ;*
0477  F7DD  A9 00       BINGCR LDA #0          ; INIT POINTERS
0478  F7DF  85 30              STA BUFPNT
0479  F7E1  85 2E              STA SAVPNT
0480  F7E3  85 36              STA BYTCNT
0481  F7E5              ;
0482  F7E5  A9 BB              LDA #256-TOPWRT
0483  F7E7  85 34              STA GCRPNT     ; START SAVING GCR HERE
0484  F7E9              ;
0485  F7E9  85 50              STA GCRFLG     ; BUFFER CONVERTED FLAG
0486  F7EB              ;
0487  F7EB  A5 31              LDA BUFPNT+1   ; SAVE BUFFER POINTER
0488  F7ED  85 2F              STA SAVPNT+1
0489  F7EF              ;
0490  F7EF  A9 01              LDA #>OVRBUF   ; POINT AT OVERFLOW
0491  F7F1  85 31              STA BUFPNT+1
0492  F7F3              ;
0493  F7F3  A5 47              LDA DBID       ; STORE DATA BLOCK ID
0494  F7F5  85 52              STA BTAB       ; AND NEXT 3 DATA BYTES
0495  F7F7              ;
0496  F7F7  A4 36              LDY BYTCNT
0497  F7F9              ;
0498  F7F9  B1 2E              LDA (SAVPNT),Y
0499  F7FB  85 53              STA BTAB+1
0500  F7FD  C8                 INY
0501  F7FE              ;
0502  F7FE  B1 2E              LDA (SAVPNT),Y
0503  F800  85 54              STA BTAB+2
0504  F802  C8                 INY
0505  F803              ;
0506  F803  B1 2E              LDA (SAVPNT),Y
0507  F805  85 55              STA BTAB+3
0508  F807  C8                 INY
0509  F808              ;
0510  F808  84 36       BING07 STY BYTCNT      ; NEXT BYTE TO GET
0511  F80A              ;
```

```
LINE# LOC   CODE        LINE


0512  F80A  20 1E F7           JSR PUT4BG     ; CONVERT AND STORE
0513  F80D              ;
0514  F80D  A4 36              LDY BYTCNT
0515  F80F              ;
0516  F80F  B1 2E              LDA (SAVPNT),Y
0517  F811  85 52              STA BTAB
0518  F813  C8                 INY
0519  F814  F0 11              BEQ BING20
0520  F816              ;
0521  F816  B1 2E              LDA (SAVPNT),Y
0522  F818  85 53              STA BTAB+1
0523  F81A  C8                 INY
0524  F81B              ;
0525  F81B  B1 2E              LDA (SAVPNT),Y
0526  F81D  85 54              STA BTAB+2
0527  F81F  C8                 INY
0528  F820              ;
0529  F820  B1 2E              LDA (SAVPNT),Y
0530  F822  85 55              STA BTAB+3
0531  F824  C8                 INY
0532  F825              ;
0533  F825  D0 E1              BNE BING07     ; JMP
0534  F827              ;
0535  F827              ;
0536  F827  A5 3A       BING20 LDA CHKSUM      ; STORE CHKSUM
0537  F829  85 53              STA BTAB+1
0538  F82B              ;
0539  F82B  A9 00              LDA #0         ; STORE 0 OFF BYTE
0540  F82D  85 54              STA BTAB+2
0541  F82F  85 55              STA BTAB+3
0542  F831              ;
0543  F831  4C 1E F7           JMP PUT4BG     ; CONVERT AND STORE AND RETURN
0544  F834              ;
0545  F834              ;
0546  F834              ;
0547  F834              ;.END
0547  F834              ;
0548  F834                     .LIB LCCGCRBIN
```

```
LINE# LOC   CODE         LINE


0550  F834               MASK1=$F8
0551  F834               MASK2=$07
0552  F834               MASK2X=$C0
0553  F834               MASK3=$3E
0554  F834               MASK4=$01
0555  F834               MASK4X=$F0
0556  F834               MASK5=$0F
0557  F834               MASK5X=$80
0558  F834               MASK6=$7C
0559  F834               MASK7=$03
0560  F834               MASK7X=$E0
0561  F834               MASK8=$1F
0562  F834               ;
0563  F834               ;
0564  F834               ;
0565  F834               ;
0566  F834               ;
0567  F834               ; FAST GCR TO BINARY CONVERSION
0568  F834               ;
0569  F834               ;
0570  F834  A4 34        GET4GB LDY GCRPNT
0571  F836               ;
0572  F836  B1 30               LDA (BUFPNT),Y
0573  F838  29 F8               AND #MASK1
0574  F83A  4A                  LSR A
0575  F83B  4A                  LSR A
0576  F83C  4A                  LSR A
0577  F83D  85 56               STA GTAB       ; HI NIBBLE
0578  F83F               ;
0579  F83F  B1 30               LDA (BUFPNT),Y
0580  F841  29 07               AND #MASK2
0581  F843  0A                  ASL A
0582  F844  0A                  ASL A
0583  F845  85 57               STA GTAB+1
0584  F847  C8                  INY            ; NEXT BYTE
0585  F848  D0 06               BNE XX05       ; TEST FOR NEXT BUFFER
0586  F84A  A5 4E               LDA NXTBF
0587  F84C  85 31               STA BUFPNT+1
0588  F84E  A4 4F               LDY NXTPNT
0589  F850               ;
0590  F850  B1 30        XX05   LDA (BUFPNT),Y
0591  F852  29 C0               AND #$C0
0592  F854  2A                  ROL A
0593  F855  2A                  ROL A
0594  F856  2A                  ROL A
0595  F857  05 57               ORA GTAB+1
0596  F859  85 57               STA GTAB+1
0597  F85B               ;
0598  F85B  B1 30               LDA (BUFPNT),Y
0599  F85D  29 3E               AND #MASK3
0600  F85F  4A                  LSR A
0601  F860  85 58               STA GTAB+2
0602  F862               ;
0603  F862  B1 30               LDA (BUFPNT),Y
0604  F864  29 01               AND #MASK4
```

```
LINE# LOC   CODE         LINE


0605  F866  0A                  ASL A
0606  F867  0A                  ASL A
0607  F868  0A                  ASL A
0608  F869  0A                  ASL A
0609  F86A  85 59               STA GTAB+3
0610  F86C               ;
0611  F86C  C8                  INY             ;NEXT
0612  F86D               ;
0613  F86D  B1 30               LDA (BUFPNT),Y
0614  F86F  29 F0               AND #MASK4X
0615  F871  4A                  LSR A
0616  F872  4A                  LSR A
0617  F873  4A                  LSR A
0618  F874  4A                  LSR A
0619  F875  05 59               ORA GTAB+3
0620  F877  85 59               STA GTAB+3
0621  F879               ;
0622  F879  B1 30               LDA (BUFPNT),Y
0623  F87B  29 0F               AND #MASK5
0624  F87D  0A                  ASL A
0625  F87E  85 5A               STA GTAB+4
0626  F880               ;
0627  F880  C8                  INY             ;NEXT BYTE
0628  F881               ;
0629  F881  B1 30               LDA (BUFPNT),Y
0630  F883  29 80               AND #MASK5X
0631  F885  18                  CLC
0632  F886  2A                  ROL A
0633  F887  2A                  ROL A
0634  F888  29 01               AND #1
0635  F88A  05 5A               ORA GTAB+4
0636  F88C  85 5A               STA GTAB+4
0637  F88E               ;
0638  F88E  B1 30               LDA (BUFPNT),Y
0639  F890  29 7C               AND #MASK6
0640  F892  4A                  LSR A
0641  F893  4A                  LSR A
0642  F894  85 5B               STA GTAB+5
0643  F896               ;
0644  F896  B1 30               LDA (BUFPNT),Y
0645  F898  29 03               AND #MASK7
0646  F89A  0A                  ASL A
0647  F89B  0A                  ASL A
0648  F89C  0A                  ASL A
0649  F89D  85 5C               STA GTAB+6
0650  F89F               ;
0651  F89F  C8                  INY             ; TEST FOR OVERFLOW DURING WRITE
                                                  TO BINARY CONVERSION
0652  F8A0               ;
0653  F8A0  D0 06               BNE XX06
0654  F8A2  A5 4E               LDA NXTBF
0655  F8A4  85 31               STA BUFPNT+1
0656  F8A6  A4 4F               LDY NXTPNT
0657  F8A8               ;
0658  F8A8  B1 30        XX06   LDA (BUFPNT),Y
0659  F8AA  29 E0               AND #MASK7X
```

```
LINE# LOC   CODE         LINE


0660  F8AC  2A                  ROL A
0661  F8AD  2A                  ROL A
0662  F8AE  2A                  ROL A
0663  F8AF  2A                  ROL A
0664  F8B0  05 5C               ORA GTAB+6
0665  F8B2  85 5C               STA GTAB+6
0666  F8B4               ;
0667  F8B4  B1 30               LDA (BUFPNT),Y
0668  F8B6  29 1F               AND #MASK8
0669  F8B8  85 5D               STA GTAB+7
0670  F8BA  C8                  INY
0671  F8BB               ;
0672  F8BB  84 34               STY GCRPNT
0673  F8BD               ;
0674  F8BD               ;
0675  F8BD  A6 56               LDX GTAB
0676  F8BF  BD EE F8            LDA GCRHI,X
0677  F8C2  A6 57               LDX GTAB+1
0678  F8C4  1D 0E F9            ORA GCRLO,X
0679  F8C7  85 52               STA BTAB
0680  F8C9               ;
0681  F8C9  A6 58               LDX GTAB+2
0682  F8CB  BD EE F8            LDA GCRHI,X
0683  F8CE  A6 59               LDX GTAB+3
0684  F8D0  1D 0E F9            ORA GCRLO,X
0685  F8D3  85 53               STA BTAB+1
0686  F8D5               ;
0687  F8D5  A6 5A               LDX GTAB+4
0688  F8D7  BD EE F8            LDA GCRHI,X
0689  F8DA  A6 5B               LDX GTAB+5
0690  F8DC  1D 0E F9            ORA GCRLO,X
0691  F8DF  85 54               STA BTAB+2
0692  F8E1               ;
0693  F8E1  A6 5C               LDX GTAB+6
0694  F8E3  BD EE F8            LDA GCRHI,X
0695  F8E6  A6 5D               LDX GTAB+7
0696  F8E8  1D 0E F9            ORA GCRLO,X
0697  F8EB  85 55               STA BTAB+3
0698  F8ED               ;
0699  F8ED  60                  RTS
0700  F8EE               ;
0701  F8EE               ;
0702  F8EE               ; TABLE FOR GCR TO BINARY CONVERSION
0703  F8EE  FF           GCRHI  .BYTE $FF        ;ERROR
0704  F8EF  FF                  .BYTE $FF       ;ERROR
0705  F8F0  FF                  .BYTE $FF       ;ERROR
0706  F8F1  FF                  .BYTE $FF       ;ERROR
0707  F8F2  FF                  .BYTE $FF       ;ERROR
0708  F8F3  FF                  .BYTE $FF       ;ERROR
0709  F8F4  FF                  .BYTE $FF       ;ERROR
0710  F8F5  FF                  .BYTE $FF       ;ERROR
0711  F8F6  FF                  .BYTE $FF       ;ERROR
0712  F8F7  80                  .BYTE $80
0713  F8F8  00                  .BYTE $00
0714  F8F9  10                  .BYTE $10
```

```
LINE# LOC   CODE         LINE


0715  F8FA  FF                   .BYTE $FF       ;ERROR
0716  F8FB  C0                   .BYTE $C0
0717  F8FC  40                   .BYTE $40
0718  F8FD  50                   .BYTE $50
0719  F8FE  FF                   .BYTE $FF       ;ERROR
0720  F8FF  FF                   .BYTE $FF       ;ERROR
0721  F900  20                   .BYTE $20
0722  F901  30                   .BYTE $30
0723  F902  FF                   .BYTE $FF       ;ERROR
0724  F903  F0                   .BYTE $F0
0725  F904  60                   .BYTE $60
0726  F905  70                   .BYTE $70
0727  F906  FF                   .BYTE $FF       ;ERROR
0728  F907  90                   .BYTE $90
0729  F908  A0                   .BYTE $A0
0730  F909  B0                   .BYTE $B0
0731  F90A  FF                   .BYTE $FF       ;ERROR
0732  F90B  D0                   .BYTE $D0
0733  F90C  E0                   .BYTE $E0
0734  F90D  FF                   .BYTE $FF       ;ERROR
0735  F90E  FF           GCRLO   .BYTE $FF        ;ERROR
0736  F90F  FF                   .BYTE $FF       ;ERROR
0737  F910  FF                   .BYTE $FF       ;ERROR
0738  F911  FF                   .BYTE $FF       ;ERROR
0739  F912  FF                   .BYTE $FF       ;ERROR
0740  F913  FF                   .BYTE $FF       ;ERROR
0741  F914  FF                   .BYTE $FF       ;ERROR
0742  F915  FF                   .BYTE $FF       ;ERROR
0743  F916  FF                   .BYTE $FF       ;ERROR
0744  F917  08                   .BYTE $08
0745  F918  00                   .BYTE $00
0746  F919  01                   .BYTE $01
0747  F91A  FF                   .BYTE $FF       ;ERROR
0748  F91B  0C                   .BYTE $0C
0749  F91C  04                   .BYTE $04
0750  F91D  05                   .BYTE $05
0751  F91E  FF                   .BYTE $FF       ;ERROR
0752  F91F  FF                   .BYTE $FF       ;ERROR
0753  F920  02                   .BYTE $02
0754  F921  03                   .BYTE $03
0755  F922  FF                   .BYTE $FF       ;ERROR
0756  F923  0F                   .BYTE $0F
0757  F924  06                   .BYTE $06
0758  F925  07                   .BYTE $07
0759  F926  FF                   .BYTE $FF       ;ERROR
0760  F927  09                   .BYTE $09
0761  F928  0A                   .BYTE $0A
0762  F929  0B                   .BYTE $0B
0763  F92A  FF                   .BYTE $FF       ;ERROR
0764  F92B  0D                   .BYTE $0D
0765  F92C  0E                   .BYTE $0E
0766  F92D  FF                   .BYTE $FF       ;ERROR
0767  F92E               ;
0768  F92E               ; CONVERT BUFFER FROM GCR TO BINARY
0769  F92E  A9 00        GCRBIN LDA #0            ; SETUP POINTERS
```

```
LINE# LOC   CODE        LINE


0770  F930  85 34                  STA GCRPNT
0771  F932  85 2E                  STA SAVPNT
0772  F934  85 36                  STA BYTCNT
0773  F936              ;
0774  F936  A9 01                  LDA #>OVRBUF
0775  F938  85 4E                  STA NXTBF
0776  F93A              ;
0777  F93A  A9 BA                  LDA #255-TOPRD
0778  F93C  85 4F                  STA NXTPNT
0779  F93E              ;
0780  F93E  A5 31                  LDA BUFPNT+1
0781  F940  85 2F                  STA SAVPNT+1
0782  F942              ;
0783  F942  20 34 F8               JSR GET4GB
0784  F945              ;
0785  F945  A5 52                  LDA BTAB
0786  F947  85 38                  STA BID        ; GET HEADER ID
0787  F949              ;
0788  F949  A4 36                  LDY BYTCNT
0789  F94B  A5 53                  LDA BTAB+1
0790  F94D  91 2E                  STA (SAVPNT),Y
0791  F94F  C8                     INY
0792  F950              ;
0793  F950  A5 54                  LDA BTAB+2
0794  F952  91 2E                  STA (SAVPNT),Y
0795  F954  C8                     INY
0796  F955              ;
0797  F955  A5 55                  LDA BTAB+3
0798  F957  91 2E                  STA (SAVPNT),Y
0799  F959  C8                     INY
0800  F95A              ;
0801  F95A  84 36       GCRB10 STY BYTCNT
0802  F95C              ;
0803  F95C  20 34 F8               JSR GET4GB
0804  F95F              ;
0805  F95F  A4 36                  LDY BYTCNT
0806  F961              ;
0807  F961  A5 52                  LDA BTAB
0808  F963  91 2E                  STA (SAVPNT),Y
0809  F965  C8                     INY
0810  F966  F0 11                  BEQ GCRB20     ; TEST IF DONE YET
0811  F968              ;
0812  F968  A5 53                  LDA BTAB+1
0813  F96A  91 2E                  STA (SAVPNT),Y
0814  F96C  C8                     INY
0815  F96D              ;
0816  F96D  A5 54                  LDA BTAB+2
0817  F96F  91 2E                  STA (SAVPNT),Y
0818  F971  C8                     INY
0819  F972              ;
0820  F972  A5 55                  LDA BTAB+3
0821  F974  91 2E                  STA (SAVPNT),Y
0822  F976  C8                     INY
0823  F977              ;
0824  F977  D0 E1                  BNE GCRB10     ; JMP
```

```
LINE# LOC   CODE         LINE


0825  F979               ;
0826  F979               GCRB20
0827  F979  A5 53               LDA BTAB+1
0828  F97B  85 3A               STA CHKSUM
0829  F97D  A5 2F               LDA SAVPNT+1   ; RESTORE BUFFER POINTER
0830  F97F  85 31               STA BUFPNT+1
0831  F981               ;
0832  F981  60                  RTS
0833  F982               ;
0834  F982               ;
0835  F982               ;
0836  F982               ;.END
0836  F982               ;
0837  F982                      .LIB LCCCONHDR
```

```
LINE# LOC   CODE         LINE


0839  F982               ;
0840  F982               ;
0841  F982               ;
0842  F982               ; *CONHDR
0843  F982               ;
0844  F982               ; CONVERT HEADER
0845  F982               ; INTO GCR SEARCH IMAGE
0846  F982               ; AND PLACE IN STAB
0847  F982               ;
0848  F982               ; IMAGE CONTAINS :
0849  F982               ;
0850  F982               ; 00 ID ID TR SC CS HBID
0851  F982               ;
0852  F982               ;
0853  F982  A5 31        CONHDR LDA BUFPNT+1    ;SAVE BUFFER POINTER
0854  F984  85 2F               STA SAVPNT+1
0855  F986               ;
0856  F986  A9 00               LDA #>STAB
0857  F988  85 31               STA BUFPNT+1
0858  F98A               ;
0859  F98A  A9 24               LDA #<STAB
0860  F98C  85 34               STA GCRPNT
0861  F98E               ;
0862  F98E  A5 39               LDA HBID
0863  F990  85 52               STA BTAB
0864  F992               ;
0865  F992  A5 1A               LDA HEADER+4
0866  F994  85 53               STA BTAB+1
0867  F996               ;
0868  F996  A5 19               LDA HEADER+3
0869  F998  85 54               STA BTAB+2
0870  F99A               ;
0871  F99A  A5 18               LDA HEADER+2
0872  F99C  85 55               STA BTAB+3
0873  F99E               ;
0874  F99E  20 1E F7            JSR PUT4BG
0875  F9A1               ;
0876  F9A1  A5 17               LDA HEADER+1
0877  F9A3  85 52               STA BTAB
0878  F9A5               ;
0879  F9A5  A5 16               LDA HEADER
0880  F9A7  85 53               STA BTAB+1
0881  F9A9               ;
0882  F9A9  A9 00               LDA #0
0883  F9AB  85 54               STA BTAB+2
0884  F9AD  85 55               STA BTAB+3
0885  F9AF               ;
0886  F9AF  20 1E F7            JSR PUT4BG
0887  F9B2               ;
0888  F9B2  A5 2F               LDA SAVPNT+1   ;RESTORE BUFFER POINTER
0889  F9B4  85 31               STA BUFPNT+1
0890  F9B6               ;
0891  F9B6  60                  RTS
0892  F9B7               ;
0893  F9B7               ;
```

```
LINE# LOC   CODE        LINE


0894  F9B7              ;.END
0894  F9B7              ;
0895  F9B7                     .LIB LCCUTIL
```

```
LINE# LOC   CODE        LINE


0897  F9B7              ;
0898  F9B7              ;
0899  F9B7              ; * UTILITY ROUTINES
0900  F9B7              ;
0901  F9B7              ;
0902  F9B7  A4 3F       ERRR   LDY JOBN         ; RETURN JOB CODE
0903  F9B9  99 00 00          STA JOBS,Y
0904  F9BC              ;
0905  F9BC  A5 50              LDA GCRFLG      ; TEST IF BUFFER LEFT GCR
0906  F9BE  F0 03              BEQ ERRR10      ; NO
0907  F9C0              ;
0908  F9C0  20 40 F6           JSR WTOBIN      ; CONVERT BACK TO BINARY
0909  F9C3              ;
0910  F9C3              ERRR10
0911  F9C3  20 DD F9           JSR TRNOFF      ; START TIMEOUT ON DRIVE
0912  F9C6              ;
0913  F9C6  A6 49              LDX SAVSP
0914  F9C8  9A                 TXS             ; RESET STACK POINTER
0915  F9C9              ;
0916  F9C9  4C 0C F3           JMP TOP         ; BACK TO THE TOP
0917  F9CC              ;
0918  F9CC              ;
0919  F9CC              ; SWITCH DRIVE MOTOR ON
0920  F9CC  A9 A0       TURNON LDA #$A0         ; TURN ON DRIVE
0921  F9CE                                     ; DRVST=ACEL AND ON
0922  F9CE  85 20              STA DRVST
0923  F9D0              ;
0924  F9D0              ;
0925  F9D0  AD 00 1C           LDA DSKCNT      ; TURN MOTOR ON AND SELECT DRIVE
0926  F9D3  09 04              ORA #$04        ; TURN MOTOR ON
0927  F9D5  8D 00 1C           STA DSKCNT
0928  F9D8              ;
0929  F9D8  A9 3C              LDA #60         ; DELAY 1.5 SEC
0930  F9DA  85 48              STA ACLTIM
0931  F9DC              ;
0932  F9DC  60                 RTS
0933  F9DD              ;
0934  F9DD              ;
0935  F9DD              ;
0936  F9DD  A6 3E       TRNOFF LDX CDRIVE       ; START TIME OUT OF CURRENT DRIVE
0937  F9DF  A5 20              LDA DRVST       ; STATUS=TIMEOUT
0938  F9E1  09 10              ORA #$10
0939  F9E3  85 20              STA DRVST
0940  F9E5              ;
0941  F9E5  A9 FF              LDA #255        ; 255*.025S TIME OUT
0942  F9E7  85 48              STA ACLTIM
0943  F9E9              ;
0944  F9E9  60                 RTS
0945  F9EA              ;
0946  F9EA              ;
0947  F9EA              ;
0948  F9EA              ;.END
0948  F9EA              ;
0949  F9EA                     .LIB LCCEND
```

```
LINE# LOC   CODE         LINE


0951  F9EA               ;
0952  F9EA               ;
0953  F9EA               ;
0954  F9EA               ; MOTOR AND STEPPER CONTROL
0955  F9EA               ;
0956  F9EA               ;
0957  F9EA               ; IRQ INTO CONTROLLER EVERY 15 MS
0958  F9EA               END
0959  F9EA  AD 07 1C            LDA T1HL2      ; SET IRQ TIMER
0960  F9ED  8D 05 1C            STA T1HC2
0961  F9F0               ;
0962  F9F0  AD 00 1C            LDA DSKCNT
0963  F9F3               ;
0964  F9F3               END001
0965  F9F3  29 10               AND #$10       ; TEST WRITE PROCTECT
0966  F9F5  C5 1E               CMP LWPT
0967  F9F7  85 1E               STA LWPT       ; CHANGE ?
0968  F9F9  F0 04               BEQ END002     ; NO
0969  F9FB               ;
0970  F9FB  A9 01               LDA #1         ; YES, SET FLAG
0971  F9FD  85 1C               STA WPSW
0972  F9FF               ;
0973  F9FF  AD FE 02     END002 LDA PHASE       ; TEST FOR PHASE OFFSET
0974  FA02  F0 15               BEQ END40
0975  FA04               ;
0976  FA04  C9 02               CMP #2
0977  FA06  D0 07               BNE END003
0978  FA08               ;
0979  FA08  A9 00               LDA #0         ; PHASE <-- 0
0980  FA0A  8D FE 02            STA PHASE
0981  FA0D  F0 0A               BEQ END40
0982  FA0F               ;
0983  FA0F  85 4A        END003 STA STEPS
0984  FA11  A9 02               LDA #2         ; PHASE <-- 2
0985  FA13  8D FE 02            STA PHASE
0986  FA16  4C 7C FA            JMP DOSTEP
0987  FA19               ;
0988  FA19  A6 3E        END40  LDX CDRIVE      ; WORK ON ACTIVE DRIVE ONLY
0989  FA1B  30 07               BMI END33X     ; NO ACTIVE DRIVE
0990  FA1D               ;
0991  FA1D  A5 20               LDA DRVST      ; TEST IF MOTOR ON
0992  FA1F  A8                  TAY
0993  FA20  C9 20               CMP #$20       ; TEST IF ANYTHING TO DO
0994  FA22  D0 03               BNE END10      ; SOMETHING HERE
0995  FA24               ;
0996  FA24  4C 0C FB     END33X JMP END33       ; MOTOR JUST RUNNING
0997  FA27               ;
0998  FA27  C6 48        END10  DEC ACLTIM      ; DEC TIMER
0999  FA29  D0 1D               BNE END30
1000  FA2B               ;
1001  FA2B  98                  TYA            ; TEST IF ACEL
1002  FA2C  10 04               BPL END20
1003  FA2E               ;
1004  FA2E               ;
1005  FA2E  29 7F               AND #$7F       ; OVER, CLEAR ACEL BIT
```

```
LINE# LOC   CODE        LINE


1006  FA30  85 20              STA DRVST
1007  FA32              ;
1008  FA32  29 10       END20  AND #$10        ; TEST IF TIME OUT STATE
1009  FA34  F0 12              BEQ END30
1010  FA36              ;
1011  FA36  AD 00 1C           LDA DSKCNT
1012  FA39  29 FB              AND #$FF-$04   ; TURNOFF MOTOR
1013  FA3B  8D 00 1C           STA DSKCNT
1014  FA3E              ;
1015  FA3E              ;
1016  FA3E  A9 FF              LDA #$FF       ; NO ACTIVE DRIVE NOW
1017  FA40  85 3E              STA CDRIVE
1018  FA42              ;
1019  FA42  A9 00              LDA #0         ; DRIVE INACTIVE
1020  FA44  85 20              STA DRVST      ; CLEAR ON BIT AND TIMOUT
1021  FA46  F0 DC              BEQ END33X
1022  FA48              ;
1023  FA48  98          END30  TYA             ; TEST IF STEP NEEDED
1024  FA49  29 40              AND #$40
1025  FA4B  D0 03              BNE END30X     ; STEPPING
1026  FA4D              ;
1027  FA4D  4C 0C FB           JMP END33
1028  FA50              ;
1029  FA50              ;
1030  FA50  6C 62 00    END30X JMP (NXTST)     ; GOTO PROPER STEPPER STATE
1031  FA53              ;
1032  FA53              ; STEPPER MOTOR CONTROL
1033  FA53  A5 4A       INACT  LDA STEPS       ; GET ABS(STEPS)
1034  FA55  10 05              BPL INAC10
1035  FA57              ;
1036  FA57  49 FF              EOR #$FF
1037  FA59  18                 CLC
1038  FA5A  69 01              ADC #1
1039  FA5C              ;
1040  FA5C  C5 64       INAC10 CMP MINSTP      ; TEST IF WE CAN ACCEL
1041  FA5E  B0 0A              BCS INAC20     ; TOO SMALL
1042  FA60              ;
1043  FA60  A9 89              LDA #<SHORT    ; SHORT STEP MODE
1044  FA62  85 62              STA NXTST
1045  FA64  A9 FA              LDA #>SHORT
1046  FA66  85 63              STA NXTST+1
1047  FA68  D0 12              BNE DOSTEP
1048  FA6A              ;
1049  FA6A              INAC20                 ; CALC THE # OF RUN STEPS
1050  FA6A  E5 5E              SBC AS
1051  FA6C  E5 5E              SBC AS
1052  FA6E  85 61              STA RSTEPS
1053  FA70              ;
1054  FA70  A5 5E              LDA AS
1055  FA72  85 60              STA ACLSTP     ; SET # OF ACCEL STEPS
1056  FA74  A9 C9              LDA #<SSACL
1057  FA76  85 62              STA NXTST
1058  FA78  A9 FA              LDA #>SSACL
1059  FA7A  85 63              STA NXTST+1
1060  FA7C              ;
```

```
LINE# LOC   CODE        LINE


1061  FA7C  A5 4A       DOSTEP LDA STEPS
1062  FA7E  10 31              BPL STPIN
1063  FA80  E6 4A       STPOUT INC STEPS
1064  FA82  AE 00 1C           LDX DSKCNT
1065  FA85  CA                 DEX
1066  FA86  4C B7 FA           JMP STP
1067  FA89                     ;
1068  FA89                     ; SLOW STEPPING MODE
1069  FA89  A5 4A       SHORT  LDA STEPS        ; STEP END ?
1070  FA8B  D0 EF              BNE DOSTEP      ; NO
1071  FA8D                     ;
1072  FA8D  A9 9C              LDA #<SETLE     ; SETTLE
1073  FA8F  85 62              STA NXTST
1074  FA91  A9 FA              LDA #>SETLE
1075  FA93  85 63              STA NXTST+1
1076  FA95  A9 05              LDA #5          ; SETTLE TIME (5*8=40MS)
1077  FA97  85 60              STA ACLSTP
1078  FA99  4C 0C FB           JMP END33
1079  FA9C                     ;
1080  FA9C                     ; END OF STEPPING
1081  FA9C  C6 60       SETLE  DEC ACLSTP       ; SETTLE END ?
1082  FA9E  D0 6C              BNE END33       ; NO
1083  FAA0                     ;
1084  FAA0  A5 20              LDA DRVST
1085  FAA2  29 BF              AND #$FF-$40
1086  FAA4  85 20              STA DRVST
1087  FAA6                     ;
1088  FAA6  A9 53              LDA #<INACT
1089  FAA8  85 62              STA NXTST
1090  FAAA  A9 FA              LDA #>INACT
1091  FAAC  85 63              STA NXTST+1
1092  FAAE  4C 0C FB           JMP END33
1093  FAB1                     ;
1094  FAB1  C6 4A       STPIN  DEC STEPS
1095  FAB3  AE 00 1C           LDX DSKCNT
1096  FAB6  E8                 INX
1097  FAB7                     ;
1098  FAB7  8A          STP    TXA
1099  FAB8  29 03              AND #3
1100  FABA  85 4B              STA TMP
1101  FABC  AD 00 1C           LDA DSKCNT
1102  FABF  29 FC              AND #$FF-$03    ; MASK OUT OLD
1103  FAC1  05 4B              ORA TMP
1104  FAC3  8D 00 1C           STA DSKCNT
1105  FAC6  4C 0C FB           JMP END33
1106  FAC9                     ;
1107  FAC9                     ; START STEPPER MOTOR
1108  FAC9              SSACL                   ; SUB ACEL FACTOR
1109  FAC9  38                 SEC
1110  FACA  AD 07 1C           LDA T1HL2
1111  FACD  E5 5F              SBC AF
1112  FACF  8D 05 1C           STA T1HC2
1113  FAD2                     ;
1114  FAD2  C6 60              DEC ACLSTP
1115  FAD4  D0 0C              BNE SSA10
```

```
LINE# LOC   CODE        LINE


1116  FAD6              ;
1117  FAD6  A5 5E              LDA AS
1118  FAD8  85 60              STA ACLSTP
1119  FADA              ;
1120  FADA  A9 E5              LDA #<SSRUN
1121  FADC  85 62              STA NXTST
1122  FADE  A9 FA              LDA #>SSRUN
1123  FAE0  85 63              STA NXTST+1
1124  FAE2              ;
1125  FAE2  4C 7C FA    SSA10  JMP DOSTEP
1126  FAE5              ;
1127  FAE5              ; FAST STEPPING
1128  FAE5  C6 61       SSRUN  DEC RSTEPS
1129  FAE7  D0 F9              BNE SSA10
1130  FAE9              ;
1131  FAE9  A9 F3              LDA #<SSDEC
1132  FAEB  85 62              STA NXTST
1133  FAED  A9 FA              LDA #>SSDEC
1134  FAEF  85 63              STA NXTST+1
1135  FAF1  D0 EF              BNE SSA10
1136  FAF3              ;
1137  FAF3              SSDEC                  ; DECEL
1138  FAF3  AD 07 1C           LDA T1HL2
1139  FAF6  18                 CLC
1140  FAF7  65 5F              ADC AF
1141  FAF9  8D 05 1C           STA T1HC2
1142  FAFC              ;
1143  FAFC  C6 60              DEC ACLSTP
1144  FAFE  D0 E2              BNE SSA10
1145  FB00              ;
1146  FB00  A9 9C              LDA #<SETLE    ; GOTO SETTLE MODE
1147  FB02  85 62              STA NXTST
1148  FB04  A9 FA              LDA #>SETLE
1149  FB06  85 63              STA NXTST+1
1150  FB08              ;
1151  FB08  A9 05              LDA #5
1152  FB0A  85 60              STA ACLSTP     ; SETTLE TIMER
1153  FB0C              ;
1154  FB0C              ;
1155  FB0C  AD 0C 1C    END33  LDA PCR2        ; DISABLE S.O. TO 6502
1156  FB0F  29 FD              AND #$FF-$02
1157  FB11  8D 0C 1C           STA PCR2
1158  FB14              ;
1159  FB14  60                 RTS
1160  FB15              ;
1161  FB15              ;
1162  FB15              ;
1163  FB15              ;.END
1163  FB15              ;
1164  FB15                     .LIB LCCFMT1
```

```
LINE# LOC   CODE        LINE


1166  FB15              ;
1167  FB15              FMTVAR = $620          ; PUT FORMAT VARS IN JUMP BUFFER
1168  FB15              CNT    = FMTVAR
1169  FB15              NUM    = FMTVAR+1
1170  FB15              TRYS   = FMTVAR+3
1171  FB15              TRAL   = FMTVAR+4
1172  FB15              DTRCK  = FMTVAR+6
1173  FB15              REMDR  = FMTVAR+7
1174  FB15              SECT   = FMTVAR+8
1175  FB15              ;
1176  FB15              ;
1177  FB15              ;
1178  FB15              ;* FORMAT ROUTINE FOR LCC
1179  FB15              ;*
1180  FB15              ;*
1181  FB15              ;*
1182  FB15              ;
1183  FB15              ;
1184  FB15              ;
1185  FB15              CODE
1186  FB15  A5 51       FORMT  LDA FTNUM       ; TEST IF FORMATTING BEGUN
1187  FB17  10 2A              BPL L213       ; YES
1188  FB19              ;
1189  FB19  A6 3D              LDX DRIVE      ; NO,START UP BY BUMPING
1190  FB1B  A9 60              LDA #$60       ; STATUS=STEPPING
1191  FB1D  95 20              STA DRVST,X
1192  FB1F              ;
1193  FB1F  A9 01              LDA #1         ; DRIVE TRACK =1
1194  FB21  95 22              STA DRVTRK,X
1195  FB23  85 51              STA FTNUM      ; START ON TRACK 1
1196  FB25              ;
1197  FB25  A9 A4              LDA #256-92    ; BUMP BACK 45 STEPS
1198  FB27  85 4A              STA STEPS
1199  FB29              ;
1200  FB29  AD 00 1C           LDA DSKCNT     ; SET PHASE A
1201  FB2C  29 FC              AND #$FF-$03
1202  FB2E  8D 00 1C           STA DSKCNT
1203  FB31              ;
1204  FB31              ;
1205  FB31  A9 0A              LDA #10        ; 10 ERRORS ALLOWED
1206  FB33  8D 20 06           STA CNT
1207  FB36              ;
1208  FB36  A9 A0              LDA #<4000     ; FIRST GUESS AT TRACK SIZE
1209  FB38  8D 21 06           STA NUM
1210  FB3B  A9 0F              LDA #>4000
1211  FB3D  8D 22 06           STA NUM+1
1212  FB40              ;
1213  FB40  4C EA F9           JMP END        ; BACK TO CONTROLLER
1214  FB43              ;
1215  FB43              ;
1216  FB43              ;
1217  FB43  A0 00       L213   LDY #0          ; TEST IF ON RIGHT TRACK NUMBER
1218  FB45  D1 32              CMP (HDRPNT),Y
1219  FB47  F0 05              BEQ L214
1220  FB49              ;
```

```
LINE# LOC   CODE         LINE


1221  FB49  91 32               STA (HDRPNT),Y ; GOTO RIGHT TRACK
1222  FB4B  4C EA F9            JMP END
1223  FB4E               ;
1224  FB4E               ;
1225  FB4E  AD 00 1C     L214   LDA DSKCNT      ; TEST FOR WRITE PROTECT
1226  FB51  29 10               AND #$10
1227  FB53  D0 05               BNE TOPP       ; ITS OK
1228  FB55               ;
1229  FB55  A9 08               LDA #8         ; WRITE PROTECT ERROR
1230  FB57  4C 22 FE            JMP FMTERR
1231  FB5A               ;
1232  FB5A  20 F2 FD     TOPP   JSR SYNCLR      ; ERASE TRACK WITH SYNC
1233  FB5D               ;
1234  FB5D  20 12 FE            JSR WRTNUM     ; WRITE OUT NUM SYNCS
1235  FB60               ;
1236  FB60  A9 55               LDA #$55       ; WRITE OUT NUM NON SYNC
1237  FB62  8D 01 1C            STA DATA2
1238  FB65               ;
1239  FB65  20 12 FE            JSR WRTNUM
1240  FB68               ;
1241  FB68  20 4F FE            JSR KILL       ; KILL WRITE
1242  FB6B               ;
1243  FB6B  20 A4 F5            JSR SYNC       ; FIND SYNC
1244  FB6E               ;
1245  FB6E  A9 40               LDA #$40       ; SET TIMER MODE
1246  FB70  0D 0B 18            ORA ACR1
1247  FB73  8D 0B 18            STA ACR1
1248  FB76               ;
1249  FB76  A9 62               LDA #100-2     ; SET UP 100US TIMER
1250  FB78  8D 06 18            STA T1LL1      ; CONT MODE TIMER
1251  FB7B  A9 00               LDA #0
1252  FB7D  8D 07 18            STA T1HL1      ; HI LATCH
1253  FB80  8D 05 18            STA T1HC1      ; GET ATTENTION OF '22
1254  FB83               ;
1255  FB83               ;
1256  FB83               ;
1257  FB83  A0 00               LDY #0         ; TIME THE SYNC AND NONSYNC
                                                 SEGMENTS
1258  FB85  A2 00               LDX #0
1259  FB87               ;
1260  FB87  2C 00 1C     FWAIT  BIT DSKCNT      ; WAIT FOR SYNC
1261  FB8A  30 FB               BMI FWAIT
1262  FB8C               ;
1263  FB8C  2C 00 1C     FWAIT2 BIT DSKCNT      ; WAIT FOR NO SYNC
1264  FB8F  10 FB               BPL FWAIT2
1265  FB91               ;
1266  FB91  AD 04 18     F000   LDA T1LC1       ; RESET IFR
1267  FB94               ;
1268  FB94  2C 00 1C     F001   BIT DSKCNT      ; TIME NONSYNC AREA
1269  FB97  10 11               BPL F005       ; TIME UNTIL SYNC FOUND
1270  FB99               ;
1271  FB99  AD 0D 18            LDA IFR1       ; TEST FOR TIME OUT
1272  FB9C  0A                  ASL A
1273  FB9D  10 F5               BPL F001       ; NOT YET
1274  FB9F               ;
1275  FB9F  E8                  INX            ; .X IS LSB
```

```
LINE# LOC   CODE        LINE


1276  FBA0  D0 EF              BNE F000
1277  FBA2  C8                 INY             ; .Y IS MSB
1278  FBA3  D0 EC              BNE F000
1279  FBA5             ;
1280  FBA5  A9 02              LDA #TOLONG     ; CANT FIND SYNC
1281  FBA7  4C 22 FE           JMP FMTERR
1282  FBAA             ;
1283  FBAA  86 71       F005   STX T2           ; SAVE TIME
1284  FBAC  84 72              STY T2+1
1285  FBAE             ;
1286  FBAE  A2 00              LDX #0          ; TIME SYNC AREA
1287  FBB0  A0 00              LDY #0
1288  FBB2             ;
1289  FBB2  AD 04 18    F006   LDA T1LC1        ; RESET IFR
1290  FBB5             ;
1291  FBB5  2C 00 1C    F007   BIT DSKCNT       ; TEST FOR NO SYNC
1292  FBB8  30 11              BMI F009
1293  FBBA             ;
1294  FBBA  AD 0D 18           LDA IFR1        ; TEST FOR TIME OUT
1295  FBBD  0A                 ASL A
1296  FBBE  10 F5              BPL F007
1297  FBC0             ;
1298  FBC0  E8                 INX             ; COUNT UP ANOTHER 100US
1299  FBC1  D0 EF              BNE F006
1300  FBC3  C8                 INY             ; MSB
1301  FBC4  D0 EC              BNE F006
1302  FBC6             ;
1303  FBC6  A9 02              LDA #TOLONG     ; CANT BE THIS LONG
1304  FBC8  4C 22 FE           JMP FMTERR
1305  FBCB             ;
1306  FBCB             ;
1307  FBCB             ;* NOW CALC THE DIFFERENCE BETWEEN
1308  FBCB             ;* SYNC AND NONSYNC AND ADJUST
1309  FBCB             ;* NUM ACCORDINGLY
1310  FBCB             ;
1311  FBCB  38          F009   SEC              ; T1-T2
1312  FBCC  8A                 TXA
1313  FBCD  E5 71              SBC T2
1314  FBCF  AA                 TAX
1315  FBD0  85 70              STA T1
1316  FBD2             ;
1317  FBD2  98                 TYA
1318  FBD3  E5 72              SBC T2+1
1319  FBD5  A8                 TAY
1320  FBD6  85 71              STA T1+1
1321  FBD8             ;
1322  FBD8  10 0B              BPL F013        ; GET ABS(T1-T2)
1323  FBDA             ;
1324  FBDA  49 FF              EOR #$FF        ; MAKE +
1325  FBDC  A8                 TAY
1326  FBDD  8A                 TXA
1327  FBDE  49 FF              EOR #$FF
1328  FBE0  AA                 TAX
1329  FBE1  E8                 INX
1330  FBE2  D0 01              BNE F013
```

```
LINE# LOC   CODE        LINE


FBE4  C8                 INY
FBE5              ;
FBE5  98          F013   TYA              ; TEST IF ABS(T1-T2)<4, THAT IS
                                          CLOSE ENOUGH
FBE6  D0 04              BNE F014       ; MSB MUST BE 0
FBE8              ;
FBE8  E0 04              CPX #4         ; TEST LSB < 4
FBEA  90 18              BCC COUNT      ; ITS THERE
FBEC              ;
FBEC  06 70       F014   ASL T1           ; NUM=NUM+(DIFF/2)
FBEE  26 71              ROL T1+1
FBF0              ;
FBF0  18                 CLC
FBF1  A5 70              LDA T1
FBF3  6D 21 06           ADC NUM
FBF6  8D 21 06           STA NUM
FBF9              ;
FBF9  A5 71              LDA T1+1
FBFB  6D 22 06           ADC NUM+1
FBFE  8D 22 06           STA NUM+1
FC01              ;
FC01  4C 5A FB           JMP TOPP       ; TRY AGAIN SAM
FC04              ;
FC04              ;
FC04  A2 00       COUNT  LDX #0           ; NOW COUNT #BYTES IN DATA
                                          SEGMENT
FC06  A0 00              LDY #0
FC08  B8                 CLV
FC09              ;
FC09  AD 00 1C    CNT10  LDA DSKCNT       ; TEST FOR SYNC
FC0C  10 0E              BPL CNT20      ; FOUND SYNC
FC0E  50 F9              BVC CNT10      ; TEST IF BYTE TIME
FC10              ;
FC10  B8                 CLV            ; YES, COUNT IT
FC11  E8                 INX
FC12  D0 F5              BNE CNT10      ; KEEP COUNTING
FC14  C8                 INY
FC15  D0 F2              BNE CNT10      ; TOO MANY ?
FC17              ;
FC17  A9 03              LDA #TOMANY    ; TOMANY COUNTS
FC19  4C 22 FE           JMP FMTERR
FC1C              ;
FC1C  8A          CNT20  TXA              ; #BYTES=COUNT*2
FC1D  0A                 ASL A
FC1E  8D 25 06           STA TRAL+1
FC21              ;
FC21  98                 TYA
FC22  2A                 ROL A
FC23  8D 24 06           STA TRAL
FC26              ;
FC26  A9 BF              LDA #$FF-$40   ; CLEAR CONT MODE
FC28  2D 0B 18           AND ACR1
FC2B  8D 0B 18           STA ACR1
FC2E              ;
FC2E              ;
FC2E              ;.END
FC2E              ;
```

```
LINE# LOC   CODE         LINE

1385  FC2E                     .LIB LCCFMT2
```

```
LINE# LOC   CODE        LINE


1387  FC2E              ;
1388  FC2E              ;
1389  FC2E              ;
1390  FC2E  A9 66       DS08   LDA #$66         ; MIN BLOCK SIZE 282*5/4 -256=85
1391  FC30  8D 26 06           STA DTRCK
1392  FC33              ;
1393  FC33  A6 43              LDX SECTR       ; TOTAL BYTES= MIN*#SECORS
1394  FC35  A0 00              LDY #0
1395  FC37  98                 TYA
1396  FC38              ;
1397  FC38  18          DS10   CLC              ; MIN*#SECTORS
1398  FC39  6D 26 06           ADC DTRCK
1399  FC3C  90 01              BCC DS14
1400  FC3E              ;
1401  FC3E  C8                 INY             ;
1402  FC3F              ;
1403  FC3F  C8          DS14   INY
1404  FC40  CA                 DEX
1405  FC41  D0 F5              BNE DS10
1406  FC43              ;
1407  FC43  49 FF              EOR #$FF        ; GET 2S COMP
1408  FC45  38                 SEC
1409  FC46  69 00              ADC #0
1410  FC48              ;
1411  FC48  18                 CLC
1412  FC49  6D 25 06           ADC TRAL+1
1413  FC4C  B0 03              BCS DS15
1414  FC4E              ;
1415  FC4E  CE 24 06           DEC TRAL
1416  FC51              ;
1417  FC51  AA          DS15   TAX
1418  FC52  98                 TYA
1419  FC53  49 FF              EOR #$FF
1420  FC55  38                 SEC
1421  FC56  69 00              ADC #0
1422  FC58  18                 CLC
1423  FC59  6D 24 06           ADC TRAL
1424  FC5C              ;
1425  FC5C  10 05              BPL DS17
1426  FC5E              ;
1427  FC5E  A9 04              LDA #TOBIG      ; NOT ENOUGH SPACE
1428  FC60  4C 22 FE           JMP FMTERR
1429  FC63              ;
1430  FC63  A8          DS17   TAY
1431  FC64  8A                 TXA
1432  FC65  A2 00              LDX #0
1433  FC67              ;
1434  FC67  38          DS20   SEC
1435  FC68  E5 43              SBC SECTR
1436  FC6A  B0 03              BCS DS22
1437  FC6C              ;
1438  FC6C  88                 DEY
1439  FC6D  30 03              BMI DS30
1440  FC6F              ;
1441  FC6F  E8          DS22   INX
```

```
LINE# LOC   CODE        LINE


FC70  D0 F5                  BNE DS20
FC72              ;
FC72  8E 26 06    DS30   STX DTRCK
FC75  E0 04              CPX #GAP2      ; TEST FOR MIN SIZE
FC77  B0 05              BCS DS32
FC79              ;
FC79  A9 05              LDA #TOSMAL    ; GAP2 TO SMALL
FC7B  4C 22 FE           JMP FMTERR
FC7E              ;
FC7E  18          DS32   CLC
FC7F  65 43              ADC SECTR
FC81  8D 27 06           STA REMDR      ; GET REMAINDER SIZE
FC84              ;
FC84              ;
FC84              ;
FC84              ;
FC84              ;
FC84              ; CREATE HEADER IMAGES
FC84              ;
FC84              ;
FC84  A9 00              LDA #0
FC86  8D 28 06           STA SECT
FC89              ;
FC89  A0 00              LDY #0
FC8B  A6 3D              LDX DRIVE
FC8D              ;
FC8D  A5 39       MAK10  LDA HBID        ; HBID CS S T ID ID 0F 0F
FC8F  99 00 03           STA BUFF0,Y
FC92  C8                 INY
FC93              ;
FC93  C8                 INY            ; SKIP CHECKSUM
FC94              ;
FC94  AD 28 06           LDA SECT       ; STORE SECTOR #
FC97  99 00 03           STA BUFF0,Y
FC9A  C8                 INY
FC9B              ;
FC9B  A5 51              LDA FTNUM      ; STORE TRACK #
FC9D  99 00 03           STA BUFF0,Y
FCA0  C8                 INY
FCA1              ;
FCA1  B5 13              LDA DSKID+1,X  ; STORE ID LOW
FCA3  99 00 03           STA BUFF0,Y
FCA6  C8                 INY
FCA7              ;
FCA7  B5 12              LDA DSKID,X    ; STORE ID HI
FCA9  99 00 03           STA BUFF0,Y
FCAC  C8                 INY
FCAD              ;
FCAD  A9 0F              LDA #$0F       ; STORE GAP1 BYTES
FCAF  99 00 03           STA BUFF0,Y
FCB2  C8                 INY
FCB3  99 00 03           STA BUFF0,Y
FCB6  C8                 INY
FCB7              ;
FCB7  A9 00              LDA #0         ; CREATE CHECKSUM
```

```
LINE# LOC   CODE        LINE


1497  FCB9  59 FA 02           EOR BUFF0-6,Y
1498  FCBC  59 FB 02           EOR BUFF0-5,Y
1499  FCBF  59 FC 02           EOR BUFF0-4,Y
1500  FCC2  59 FD 02           EOR BUFF0-3,Y
1501  FCC5              ;
1502  FCC5  99 F9 02           STA BUFF0-7,Y  ; STORE CHECKSUM
1503  FCC8              ;
1504  FCC8              ;
1505  FCC8  EE 28 06           INC SECT       ; GOTO NEXT SECTOR
1506  FCCB              ;
1507  FCCB  AD 28 06           LDA SECT       ; TEST IF DONE YET
1508  FCCE  C5 43              CMP SECTR
1509  FCD0  90 BB              BCC MAK10      ; MORE TO DO
1510  FCD2              ;
1511  FCD2  98                 TYA            ; SAVE BLOCK SIZE
1512  FCD3  48                 PHA
1513  FCD4              ;
1514  FCD4              ;
1515  FCD4              ;
1516  FCD4              ; CREATE DATA BLOCK OF ZERO
1517  FCD4              ;
1518  FCD4  A2 00              LDX #0         ;.X=0
1519  FCD6  8A                 TXA
1520  FCD7              ;
1521  FCD7  9D 00 05    CRTDAT STA BUFF2,X
1522  FCDA  E8                 INX
1523  FCDB  D0 FA              BNE CRTDAT
1524  FCDD              ;
1525  FCDD              ;
1526  FCDD              ; CONVERT HEADER BLOCK TO GCR
1527  FCDD              ;
1528  FCDD  A9 03              LDA #>BUFF0
1529  FCDF  85 31              STA BUFPNT+1   ; POINT AT BUFFER
1530  FCE1              ;
1531  FCE1  20 7F FE           JSR FBTOG      ; CONVERT TO GCR WITH NO BID CHAR
1532  FCE4              ;
1533  FCE4  68                 PLA            ; RESTORE BLOCK SIZE
1534  FCE5  A8                 TAY            ; MOVE BUFFER UP 79 BYTES
1535  FCE6  88                 DEY            ; FOR I=N-1 TO
                                               0:MEM.I+69.:=MEM.I.:NEXT
1536  FCE7  20 34 FE           JSR MOVUP      ;MOVE BUF0 UP 69 BYTES
1537  FCEA              ;
1538  FCEA  20 44 FE           JSR MOVOVR     ; MOVE OVRBUF UP TO BUFFER
1539  FCED              ;
1540  FCED              ;
1541  FCED              ;
1542  FCED              ; CONVERT DATA BLOCK TO GCR
1543  FCED              ; WRITE IMAGE
1544  FCED              ;
1545  FCED              ; LEAVE IT IN OVRBUF AND BUFFER
1546  FCED              ;
1547  FCED              ;
1548  FCED  A9 05              LDA #>BUFF2    ; POINT AT BUFFER
1549  FCEF  85 31              STA BUFPNT+1
1550  FCF1              ;
1551  FCF1              ;
```

```
LINE# LOC   CODE        LINE


1552  FCF1  20 37 F6           JSR CHKBLK     ; GET BLOCK CHECKSUM
1553  FCF4  85 3A              STA CHKSUM
1554  FCF6  20 DD F7           JSR BINGCR
1555  FCF9              ;
1556  FCF9              ;
1557  FCF9              ;
1558  FCF9              ; START THE FORMAT NOW
1559  FCF9              ;
1560  FCF9              ; WRITE OUT SYNC HEADER GAP1
1561  FCF9              ; DATA BLOCK
1562  FCF9              ;
1563  FCF9              ;
1564  FCF9              ;
1565  FCF9  A9 00              LDA #0         ; INIT COUNTER
1566  FCFB  85 32              STA HDRPNT
1567  FCFD              ;
1568  FCFD  20 5D FE           JSR CLEAR
1569  FD00              ;
1570  FD00  A9 FF       WRTSYN LDA #$FF        ; WRITE SYNC
1571  FD02  8D 01 1C           STA DATA2
1572  FD05              ;
1573  FD05  A2 05              LDX #NUMSYN    ; WRITE 4 SYNC
1574  FD07              ;
1575  FD07  50 FE       WRTS10 BVC *
1576  FD09  B8                 CLV
1577  FD0A              ;
1578  FD0A  CA                 DEX
1579  FD0B  D0 FA              BNE WRTS10
1580  FD0D              ;
1581  FD0D  A2 0A              LDX #10        ; WRITE OUT HEADER
1582  FD0F  A4 32              LDY HDRPNT
1583  FD11              ;
1584  FD11  50 FE       WRTS20 BVC *
1585  FD13  B8                 CLV
1586  FD14              ;
1587  FD14  B9 00 03           LDA BUFF0,Y    ; GET HEADER DATA
1588  FD17  8D 01 1C           STA DATA2
1589  FD1A              ;
1590  FD1A  C8                 INY
1591  FD1B  CA                 DEX
1592  FD1C  D0 F3              BNE WRTS20
1593  FD1E              ;
1594  FD1E              ;
1595  FD1E              ; * WRITE OUT GAP1
1596  FD1E              ;
1597  FD1E  A2 08              LDX #GAP1-2    ; WRITE GCR BYTES (!1541 HAS 9)
1598  FD20              ;
1599  FD20  50 FE       WRTS30 BVC *
1600  FD22  B8                 CLV
1601  FD23              ;
1602  FD23  A9 55              LDA #$55
1603  FD25  8D 01 1C           STA DATA2
1604  FD28              ;
1605  FD28  CA                 DEX
1606  FD29  D0 F5              BNE WRTS30
```

```
LINE# LOC   CODE        LINE


1607  FD2B              ;
1608  FD2B              ;
1609  FD2B              ;
1610  FD2B              ; * WRITE OUT DATA BLOCK
1611  FD2B              ;
1612  FD2B  A9 FF              LDA #$FF       ; WRITE DATA BLOCK SYNC
1613  FD2D              ;
1614  FD2D  A2 05              LDX #NUMSYN
1615  FD2F              ;
1616  FD2F  50 FE       DBSYNC BVC *
1617  FD31  B8                 CLV
1618  FD32              ;
1619  FD32  8D 01 1C           STA DATA2
1620  FD35              ;
1621  FD35  CA                 DEX
1622  FD36  D0 F7              BNE DBSYNC
1623  FD38              ;
1624  FD38  A2 BB              LDX #256-TOPWRT ; WRITE OUT OVRBUF
1625  FD3A              ;
1626  FD3A  50 FE       WRTS40 BVC *
1627  FD3C  B8                 CLV
1628  FD3D              ;
1629  FD3D  BD 00 01           LDA OVRBUF,X
1630  FD40  8D 01 1C           STA DATA2
1631  FD43              ;
1632  FD43  E8                 INX
1633  FD44  D0 F4              BNE WRTS40
1634  FD46              ;
1635  FD46              ;
1636  FD46  A0 00              LDY #0
1637  FD48              ;
1638  FD48  50 FE       WRTS50 BVC *
1639  FD4A  B8                 CLV
1640  FD4B              ;
1641  FD4B  B1 30              LDA (BUFPNT),Y
1642  FD4D  8D 01 1C           STA DATA2
1643  FD50              ;
1644  FD50  C8                 INY
1645  FD51  D0 F5              BNE WRTS50
1646  FD53              ;
1647  FD53  A9 55              LDA #$55       ; WRITE GAP2(DTRCK)
1648  FD55  AE 26 06           LDX DTRCK
1649  FD58              ;
1650  FD58  50 FE       WGP2   BVC *
1651  FD5A  B8                 CLV
1652  FD5B              ;
1653  FD5B  8D 01 1C           STA DATA2
1654  FD5E  CA                 DEX
1655  FD5F  D0 F7              BNE WGP2
1656  FD61              ;
1657  FD61              ; LDX #20 ; WRITE ERASE TRAIL GAP
1658  FD61              ;WGP3 BVC *
1659  FD61              ; CLV
1660  FD61              ; DEX
1661  FD61              ; BNE WGP3
```

```
LINE# LOC   CODE        LINE


1662  FD61              ;
1663  FD61  A5 32              LDA HDRPNT     ; ADVANCE HEADER POINTER
1664  FD63  18                 CLC
1665  FD64  69 0A              ADC #10
1666  FD66  85 32              STA HDRPNT
1667  FD68              ;
1668  FD68              ;
1669  FD68              ;
1670  FD68              ; DONE WRITING SECTOR
1671  FD68              ;
1672  FD68  CE 28 06           DEC SECT       ; GO TO NEXT ON
1673  FD6B  D0 93              BNE WRTSYN     ; MORE TO DO
1674  FD6D              ;
1675  FD6D  50 FE              BVC *          ; WAIT FOR LAST ONE TO WRITE
1676  FD6F  B8                 CLV
1677  FD70              ;
1678  FD70  50 FE              BVC *
1679  FD72  B8                 CLV
1680  FD73              ;
1681  FD73  20 4F FE           JSR KILL       ; GOTO READ MODE
1682  FD76              ;
1683  FD76              ;
1684  FD76              ;
1685  FD76              ;.END
1685  FD76              ;
1686  FD76                     .LIB LCCFMT3
```

```
LINE# LOC   CODE        LINE


1688  FD76              ;
1689  FD76              ;
1690  FD76              ;
1691  FD76              ;* FORMAT DONE, NOW VERIFY IT
1692  FD76              ;
1693  FD76              ;
1694  FD76              ;
1695  FD76  A9 C8              LDA #200       ; LOOK AT 200 SYNCS
1696  FD78  8D 23 06           STA TRYS
1697  FD7B              ;
1698  FD7B  A9 00       COMP   LDA #0          ; POINTER INTO HEADERS
1699  FD7D  85 30              STA BUFPNT
1700  FD7F              ;
1701  FD7F  A9 03              LDA #>BUFF0    ;
1702  FD81  85 31              STA BUFPNT+1
1703  FD83              ;
1704  FD83  A5 43              LDA SECTR      ; SECTOR COUNTER
1705  FD85  8D 28 06           STA SECT
1706  FD88              ;
1707  FD88  20 A4 F5    CMPR10 JSR SYNC        ; FIND SYNC
1708  FD8B              ;
1709  FD8B  A2 0A              LDX #10
1710  FD8D  A0 00              LDY #0
1711  FD8F              ;
1712  FD8F  50 FE       CMPR15 BVC *           ; GET HEADER BYTE
1713  FD91  B8                 CLV
1714  FD92              ;
1715  FD92  AD 01 1C           LDA DATA2
1716  FD95  D1 30              CMP (BUFPNT),Y ; COMPARE GCR
1717  FD97              ;
1718  FD97  D0 0E              BNE CMPR20     ; ERROR
1719  FD99              ;
1720  FD99  C8                 INY
1721  FD9A  CA                 DEX
1722  FD9B  D0 F2              BNE CMPR15     ; TEST ALL BYTES
1723  FD9D              ;
1724  FD9D  18                 CLC            ; UPDATE HEADR POINTER
1725  FD9E  A5 30              LDA BUFPNT
1726  FDA0  69 0A              ADC #10
1727  FDA2  85 30              STA BUFPNT
1728  FDA4              ;
1729  FDA4  4C B1 FD           JMP TSTDAT     ; NOW TEST DATA
1730  FDA7              ;
1731  FDA7  CE 23 06    CMPR20 DEC TRYS        ; TEST IF TOO MANY ERRORS
1732  FDAA  D0 CF              BNE COMP
1733  FDAC              ;
1734  FDAC  A9 06              LDA #NOTFND    ; TOO MANY ERRORS
1735  FDAE  4C 22 FE           JMP FMTERR
1736  FDB1              ;
1737  FDB1  20 A4 F5    TSTDAT JSR SYNC        ; FIND DATA SYNC
1738  FDB4              ;
1739  FDB4  A0 BB              LDY #256-TOPWRT ;
1740  FDB6              ;
1741  FDB6  50 FE       TST05  BVC *
1742  FDB8  B8                 CLV
```

```
LINE# LOC   CODE        LINE


1743  FDB9              ;
1744  FDB9  AD 01 1C           LDA DATA2      ; COMPARE GCR
1745  FDBC  D9 00 01           CMP OVRBUF,Y
1746  FDBF              ;
1747  FDBF  D0 E6              BNE CMPR20     ; ERROR
1748  FDC1              ;
1749  FDC1  C8                 INY
1750  FDC2  D0 F2              BNE TST05      ; DO ALL OVRBUF
1751  FDC4              ;
1752  FDC4  A2 FC              LDX #255-3     ; NOW DO BUFFER, DONT TEST OFF
                                               BYTES
1753  FDC6              ;
1754  FDC6  50 FE       TST10  BVC *
1755  FDC8  B8                 CLV
1756  FDC9              ;
1757  FDC9  AD 01 1C           LDA DATA2
1758  FDCC  D9 00 05           CMP BUFF2,Y
1759  FDCF  D0 D6              BNE CMPR20
1760  FDD1              ;
1761  FDD1  C8                 INY
1762  FDD2  CA                 DEX
1763  FDD3  D0 F1              BNE TST10
1764  FDD5              ;
1765  FDD5              ;
1766  FDD5  CE 28 06           DEC SECT       ; MORE SECTORS TO TEST?
1767  FDD8  D0 AE              BNE CMPR10     ; YES
1768  FDDA              ;
1769  FDDA              ;
1770  FDDA              ; ALL SECTORS DONE OK
1771  FDDA              ;
1772  FDDA  E6 51              INC FTNUM      ; GOTO NEXT TRACK
1773  FDDC  A5 51              LDA FTNUM
1774  FDDE  C9 24              CMP #36        ; #TRACKS MAX
1775  FDE0  B0 03              BCS FMTEND
1776  FDE2              ;
1777  FDE2  4C EA F9           JMP END        ; MORE TO DO
1778  FDE5              ;
1779  FDE5              ;
1780  FDE5  A9 FF       FMTEND LDA #$FF        ; CLEAR FTNUM
1781  FDE7  85 51              STA FTNUM
1782  FDE9              ;
1783  FDE9  A9 00              LDA #$0        ; CLEAR GCR BUFFER FLAG
1784  FDEB  85 50              STA GCRFLG
1785  FDED              ;
1786  FDED  A9 01              LDA #1         ; RETURN OK CODE
1787  FDEF  4C B7 F9           JMP ERRR
1788  FDF2              ;
1789  FDF2              ;
1790  FDF2              ;
1791  FDF2              ;.END
1791  FDF2              ;
1792  FDF2                     .LIB LCCFMT4
```

```
LINE# LOC   CODE         LINE


1794  FDF2               ;
1795  FDF2               ;
1796  FDF2               ;
1797  FDF2  AD 0C 1C     SYNCLR LDA PCR2         ; WRITE ENTIRE TRACK WITH SYNC
1798  FDF5  29 1F               AND #$FF-$E0
1799  FDF7  09 C0               ORA #$C0
1800  FDF9  8D 0C 1C            STA PCR2
1801  FDFC               ;
1802  FDFC  A9 FF               LDA #$FF        ; OUTPUT MODE DDR
1803  FDFE  8D 03 1C            STA DDRA2
1804  FE01               ;
1805  FE01  8D 01 1C            STA DATA2       ; SYNC CHAR
1806  FE04               ;
1807  FE04  A2 28               LDX #$28        ; $28*256 BYTES
1808  FE06  A0 00               LDY #0
1809  FE08               ;
1810  FE08  50 FE        SYC10  BVC *
1811  FE0A  B8                  CLV
1812  FE0B               ;
1813  FE0B  88                  DEY
1814  FE0C  D0 FA               BNE SYC10
1815  FE0E               ;
1816  FE0E  CA                  DEX
1817  FE0F  D0 F7               BNE SYC10
1818  FE11               ;
1819  FE11  60                  RTS             ; LEAVE WRITE ON
1820  FE12               ;
1821  FE12               ;
1822  FE12               ;
1823  FE12  AE 21 06     WRTNUM LDX NUM          ; WRITE OUT NUM BYTES
1824  FE15  AC 22 06            LDY NUM+1
1825  FE18               ;
1826  FE18  50 FE        WRTN10 BVC *
1827  FE1A  B8                  CLV
1828  FE1B               ;
1829  FE1B  CA                  DEX
1830  FE1C  D0 FA               BNE WRTN10
1831  FE1E               ;
1832  FE1E  88                  DEY
1833  FE1F  10 F7               BPL WRTN10
1834  FE21               ;
1835  FE21  60                  RTS
1836  FE22               ;
1837  FE22               ;
1838  FE22               ;
1839  FE22  CE 20 06     FMTERR DEC CNT          ; TEST FOR RETRY
1840  FE25  F0 03               BEQ FMTE10
1841  FE27               ;
1842  FE27  4C EA F9            JMP END
1843  FE2A               ;
1844  FE2A               FMTE10
1845  FE2A               ;
1846  FE2A  A0 FF               LDY #$FF
1847  FE2C  84 51               STY FTNUM       ; CLEAR FORMAT
1848  FE2E               ;
```

```
LINE# LOC   CODE        LINE


1849  FE2E  C8                 INY
1850  FE2F  84 50              STY GCRFLG
1851  FE31              ;
1852  FE31  4C B7 F9           JMP ERRR
1853  FE34              ;
1854  FE34              ;
1855  FE34              ;
1856  FE34  B9 00 03    MOVUP  LDA BUFF0,Y      ; MOVE UP 69 BYTES
1857  FE37  99 45 03           STA BUFF0+69,Y ; MOVE FROM TOP DOWN
1858  FE3A  88                 DEY
1859  FE3B  D0 F7              BNE MOVUP
1860  FE3D              ;
1861  FE3D  AD 00 03           LDA BUFF0       ; DO LAST BYTE
1862  FE40  8D 45 03           STA BUFF0+69
1863  FE43  60                 RTS
1864  FE44              ;
1865  FE44              ;
1866  FE44              ;
1867  FE44              ;
1868  FE44  A0 44       MOVOVR LDY #68          ; MOVE OVRBUF INTO (BUFFER)
1869  FE46              ;
1870  FE46  B9 BB 01    MOVO10 LDA OVRBUF+256-TOPWRT,Y
1871  FE49  91 30              STA (BUFPNT),Y
1872  FE4B              ;
1873  FE4B  88                 DEY
1874  FE4C  10 F8              BPL MOVO10
1875  FE4E              ;
1876  FE4E  60                 RTS
1877  FE4F              ;
1878  FE4F              ;
1879  FE4F              ;
1880  FE4F  AD 0C 1C    KILL   LDA PCR2         ; DISABLE WRITE
1881  FE52  09 E0              ORA #$E0
1882  FE54  8D 0C 1C           STA PCR2
1883  FE57  A9 00              LDA #$00        ; MAKE PORT INPUT NOW
1884  FE59  8D 03 1C           STA DDRA2
1885  FE5C              ;
1886  FE5C  60                 RTS
1887  FE5D              ;
1888  FE5D              ;
1889  FE5D              ;
1890  FE5D  AD 0C 1C    CLEAR  LDA PCR2         ; ENABLE WRITE
1891  FE60  29 1F              AND #$FF-$E0
1892  FE62  09 C0              ORA #$C0
1893  FE64  8D 0C 1C           STA PCR2
1894  FE67              ;
1895  FE67  A9 FF              LDA #$FF        ; MAKE PORT AN OUTPUT
1896  FE69  8D 03 1C           STA DDRA2
1897  FE6C              ;
1898  FE6C  A9 55              LDA #$55        ; WRITE A 1F PATTERN
1899  FE6E  8D 01 1C           STA DATA2
1900  FE71              ;
1901  FE71  A2 28              LDX #$28        ; $28*256 CHARS
1902  FE73  A0 00              LDY #00
1903  FE75  50 FE       CLER10 BVC *
```

```
LINE# LOC   CODE        LINE


1904  FE77  B8                 CLV
1905  FE78  88                 DEY
1906  FE79  D0 FA              BNE CLER10
1907  FE7B              ;
1908  FE7B  CA                 DEX
1909  FE7C  D0 F7              BNE CLER10
1910  FE7E              ;
1911  FE7E  60                 RTS
1912  FE7F              ;
1913  FE7F              ;*****************************
1914  FE7F              ;*
1915  FE7F              ;*
1916  FE7F              ;* FBTOG
1917  FE7F              ;* FORMAT BINARY TO GCR CONVERSION
1918  FE7F              ;*
1919  FE7F              ;* CONVERTS BUFFER TO GCR WITH OUT HBID
1920  FE7F              ;*
1921  FE7F              ;***************************
1922  FE7F              ;
1923  FE7F  A9 00       FBTOG  LDA #0          ; POINT AT BUFFER
1924  FE81  85 30              STA BUFPNT
1925  FE83  85 2E              STA SAVPNT
1926  FE85  85 36              STA BYTCNT
1927  FE87              ;
1928  FE87  A9 BB              LDA #256-TOPWRT ; PUT GCR IN OVRFLOW BUFFER
1929  FE89  85 34              STA GCRPNT
1930  FE8B              ;
1931  FE8B  A5 31              LDA BUFPNT+1   ; SAVE BUFFER POINTER
1932  FE8D  85 2F              STA SAVPNT+1
1933  FE8F              ;
1934  FE8F  A9 01              LDA #>OVRBUF
1935  FE91  85 31              STA BUFPNT+1   ; STORE IN OVERBUF
1936  FE93              ;
1937  FE93  A4 36       FBG10  LDY BYTCNT      ; GET POINTER
1938  FE95              ;
1939  FE95  B1 2E              LDA (SAVPNT),Y
1940  FE97  85 52              STA BTAB
1941  FE99  C8                 INY
1942  FE9A              ;
1943  FE9A  B1 2E              LDA (SAVPNT),Y
1944  FE9C  85 53              STA BTAB+1
1945  FE9E  C8                 INY
1946  FE9F              ;
1947  FE9F  B1 2E              LDA (SAVPNT),Y
1948  FEA1  85 54              STA BTAB+2
1949  FEA3  C8                 INY
1950  FEA4              ;
1951  FEA4  B1 2E              LDA (SAVPNT),Y
1952  FEA6  85 55              STA BTAB+3
1953  FEA8  C8                 INY
1954  FEA9  F0 08              BEQ FBG15      ; TEST IF DONE
1955  FEAB              ;
1956  FEAB  84 36              STY BYTCNT     ; SAVE POINTER
1957  FEAD              ;
1958  FEAD  20 1E F7           JSR PUT4BG     ; CONVERT AND STORE
```

```
LINE# LOC   CODE        LINE


1959  FEB0              ;
1960  FEB0  4C 93 FE           JMP FBG10
1961  FEB3              ;
1962  FEB3  4C 1E F7    FBG15  JMP PUT4BG      ; DONE, RETURN
1963  FEB6              ;
1964  FEB6              ;.END
1964  FEB6              ;
1965  FEB6                     .LIB IRQ
```

```
LINE# LOC   CODE         LINE


1967  FEB6              ;
1968  FEB6              ;
1969  FEB6              ;
1970  FEB6  48          SYSIRQ PHA              ; SAVE .A,.X,.Y
1971  FEB7  8A                 TXA
1972  FEB8  48                 PHA
1973  FEB9  98                 TYA
1974  FEBA  48                 PHA
1975  FEBB              ;
1976  FEBB              ;
1977  FEBB  AD 0D 18           LDA IFR1       ; TEST IF ATN
1978  FEBE  29 02              AND #2
1979  FEC0  F0 03              BEQ IRQ10      ; NOT ATN
1980  FEC2              ;
1981  FEC2  20 81 E8           JSR ATNIRQ     ; HANDLE ATN REQUEST
1982  FEC5              ;
1983  FEC5              ;
1984  FEC5  AD 0D 1C    IRQ10  LDA IFR2        ; TEST IF TIMER
1985  FEC8  0A                 ASL A
1986  FEC9  10 03              BPL IRQ20      ; NOT TIMER
1987  FECB              ;
1988  FECB  20 FE F2           JSR LCC        ; GOTO CONTROLLER
1989  FECE              ;
1990  FECE  68          IRQ20  PLA              ;RESTORE .Y,.X,.A
1991  FECF  A8                 TAY
1992  FED0  68                 PLA
1993  FED1  AA                 TAX
1994  FED2  68                 PLA
1995  FED3  40                 RTI            ;AND RETURN
1996  FED4              ;
1997  FED4              ;
1998  FED4              ;
1999  FED4              ;.END
1999  FED4              ;
2000  FED4                     .LIB ROMTBLSF
```

```
LINE# LOC   CODE         LINE


2002  FED4               ;
2003  FED4               ; VARIOUS DATA TABLES
2004  FED4  12           DIRTRK .BYTE 18        ; DIRECTORY TRACK #
2005  FED5  04           BAMSIZ .BYTE 4         ; BYTES PER TRACK FOR THE BAM
2006  FED6  04           MAPOFF .BYTE 4         ; OFFSET OF BAM IN SECTOR
2007  FED7  90           DSKNAM .BYTE $90       ; OFFSET OF DISK NAME IN BAM
                                                SECTOR
2008  FED8               ;
2009  FED8               ; COMMAND SEARCH TABLE
2010  FED8  56           CMDTBL .BYTE 'V'
2011  FED9  49                  .BYTE 'I'
2012  FEDA  44                  .BYTE 'D'
2013  FEDB  4D                  .BYTE 'M'
2014  FEDC  42                  .BYTE 'B'
2015  FEDD  55                  .BYTE 'U'
2016  FEDE  50                  .BYTE 'P'
2017  FEDF  26                  .BYTE '&'
2018  FEE0  43                  .BYTE 'C'
2019  FEE1  52                  .BYTE 'R'
2020  FEE2  53                  .BYTE 'S'
2021  FEE3  4E                  .BYTE 'N'
2022  FEE4               ; VALIDATE-DIR INIT-DRIVE DUPLICATE
2023  FEE4               ; MEMORY-OP BLOCK-OP USER
2024  FEE4               ; POSITION DSKCPY UTLODR RENAME SCRATCH NEW
2025  FEE4               NCMDS  =*-CMDTBL
2026  FEE4               ; JUMP TABLE LOW
2027  FEE4  D2           CJUMPL .BYTE <VERDIR
2028  FEE5  05                  .BYTE <INTDRV
2029  FEE6  C1                  .BYTE <DUPLCT
2030  FEE7  F8                  .BYTE <MEM
2031  FEE8  1B                  .BYTE <BLOCK
2032  FEE9  5C                  .BYTE <USER
2033  FEEA  07                  .BYTE <RECORD
2034  FEEB  D1                  .BYTE <UTLODR
2035  FEEC  F0                  .BYTE <DSKCPY
2036  FEED  88                  .BYTE <RENAME
2037  FEEE  23                  .BYTE <SCRTCH
2038  FEEF  5B                  .BYTE <NEW
2039  FEF0                      *=CJUMPL+NCMDS
2040  FEF0               ; JUMP TABLE HIGH
2041  FEF0  ED           CJUMPH .BYTE >VERDIR
2042  FEF1  D0                  .BYTE >INTDRV
2043  FEF2  C8                  .BYTE >DUPLCT
2044  FEF3  CA                  .BYTE >MEM
2045  FEF4  CC                  .BYTE >BLOCK
2046  FEF5  CB                  .BYTE >USER
2047  FEF6  E2                  .BYTE >RECORD
2048  FEF7  E7                  .BYTE >UTLODR
2049  FEF8  C8                  .BYTE >DSKCPY
2050  FEF9  CA                  .BYTE >RENAME
2051  FEFA  C8                  .BYTE >SCRTCH
2052  FEFB  EE                  .BYTE >NEW
2053  FEFC                      *=CJUMPH+NCMDS
2054  FEFC               VAL=0                  ; VALIDATE (VERIFY) CMD #
2055  FEFC
2056  FEFC               ; COMMAND BIT MASKS
```

```
LINE# LOC   CODE        LINE


2057  FEFC              PCMD   =9
2058  FEFC  51                 .BYTE %01010001 ; DSKCPY (!)
2059  FEFD              STRUCT =*-PCMD         ; CMDS NOT PARSED
2060  FEFD  DD                 .BYTE %11011101 ; RENAME
2061  FEFE  1C                 .BYTE %00011100 ; SCRATCH
2062  FEFF  9E                 .BYTE %10011110 ; NEW
2063  FF00              LDCMD  =*-STRUCT       ; LOAD CMD IMAGE
2064  FF00  1C                 .BYTE %00011100 ; LOAD
2065  FF01              ; --- ---
2066  FF01              ; PGDRPGDR
2067  FF01              ; FS1 FS2

2069  FF01              ; BIT REPS: NOT PATTERN
2070  FF01              ; NOT GREATER THAN ONE FILE
2071  FF01              ; NOT DEFAULT DRIVE(S)
2072  FF01              ; REQUIRED FILENAME

2074  FF01  52          MODLST .BYTE 'R'       ;READ
2075  FF02  57                 .BYTE 'W'      ;WRITE
2076  FF03  41                 .BYTE 'A'      ;APPEND
2077  FF04  4D                 .BYTE 'M'      ;MODIFY
2078  FF05              NMODES =*-MODLST
2079  FF05              ;FILE TYPE TABLE
2080  FF05  44          TPLST  .BYTE 'D'       ;DEL
2081  FF06  53                 .BYTE 'S'      ;SEQ
2082  FF07  50                 .BYTE 'P'      ;PRG
2083  FF08  55                 .BYTE 'U'      ;USR
2084  FF09  4C                 .BYTE 'L'      ;REL
2085  FF0A              ; NAME OF FILE TYPES FOR DIRECTORY, 1ST LETTERS
2086  FF0A  44          TYPLST .BYTE 'D'
2087  FF0B  53                 .BYTE 'S'
2088  FF0C  50                 .BYTE 'P'
2089  FF0D  55                 .BYTE 'U'
2090  FF0E  52                 .BYTE 'R'
2091  FF0F              NTYPES =*-TYPLST
2092  FF0F              ; NAME OF FILE TYPES FOR DIRECTORY, 2ND LETTERS
2093  FF0F  45          TP1LST .BYTE 'E'
2094  FF10  45                 .BYTE 'E'
2095  FF11  52                 .BYTE 'R'
2096  FF12  53                 .BYTE 'S'
2097  FF13  45                 .BYTE 'E'
2098  FF14              ; NAME OF FILE TYPES FOR DIRECTORY, 3RD LETTERS
2099  FF14  4C          TP2LST .BYTE 'L'
2100  FF15  51                 .BYTE 'Q'
2101  FF16  47                 .BYTE 'G'      ;PRG
2102  FF17  52                 .BYTE 'R'
2103  FF18  4C                 .BYTE 'L'
2104  FF19              ; LED MASKS
2105  FF19  08          LEDMSK .BYTE LED0      ;DRIVE 0
2106  FF1A  00                 .BYTE LED1     ;DRIVE 1 (NOT PRESENT)
2107  FF1B              ;
2108  FF1B              ; ERROR FLAG VARS FOR BIT
2109  FF1B              ;
2110  FF1B  00          ER00   .BYTE 0
2111  FF1C  3F          ER0    .BYTE $3F
```

```
LINE# LOC   CODE        LINE


2112  FF1D  7F          ER1    .BYTE $7F
2113  FF1E  BF          ER2    .BYTE $BF
2114  FF1F  FF          ER3    .BYTE $FF
2115  FF20              ;
2116  FF20              ; TABLE: SECTORS PER TRACK
2117  FF20              NUMSEC                 ;(4) SECTORS/TRACK
2118  FF20  11                 .BYTE 17       ; TRACKS 31-35
2119  FF21  12                 .BYTE 18       ;TRACKS 25-30
2120  FF22  13                 .BYTE 19       ;TRACKS 18-24
2121  FF23  15                 .BYTE 21       ;TRACKS 01-17
2122  FF24              ; DOS VERSION IDENTIFIER
2123  FF24  41          VERNUM .BYTE FM4040    ; FORMAT TYPE
2124  FF25              ; NUMBER OF TRACK ZONES
2125  FF25  04          NZONES .BYTE 4         ; # OF ZONES
2126  FF26              MAXTRK                 ; MAXIMUM TRACK # +1
2127  FF26              ; TABLE: FIRST TRACK# OF NEXT ZONE
2128  FF26  24          TRKNUM .BYTE 36
2129  FF27  1F                 .BYTE 31
2130  FF28  19                 .BYTE 25
2131  FF29  12                 .BYTE 18
2132  FF2A              ; TABLE: BYTE SEQUENCES FOR HALF TRACK STEPS
2133  FF2A              OFFSET                 ; FOR RECOVERY
2134  FF2A  01                 .BYTE $01      ;HALF A TRACK INWARD
2135  FF2B  FF                 .BYTE $FF      ;HALF A TRACK OUTWARD
2136  FF2C  FF                 .BYTE $FF
2137  FF2D  01                 .BYTE $01
2138  FF2E  00                 .BYTE $00      ;END OF TABLE?
2139  FF2F              ;
2140  FF2F              BUFIND
2141  FF2F  03                 .BYTE $03
2142  FF30  04                 .BYTE $04
2143  FF31  05                 .BYTE $05
2144  FF32  06                 .BYTE $06
2145  FF33  07                 .BYTE $07
2146  FF34  07                 .BYTE $07
2147  FF35              ;

2149  FF35              ;.END
2149  FF35              ;
2150  FF35                     .LIB VECTOR
```

```
LINE# LOC   CODE        LINE


2152  FF35  FA          ECHKSM .BYTE $FA       ; $E - $F CHECKSUM
2153  FF36  6C 65 00    NMI    JMP (VNMI)
2154  FF39              ;
2155  FF39              PATCH                  ;$E - $F PATCH AREA
2156  FF39              ;---------------------------------------------
2157  FF39              ; IN THE 2031 901484-05 ROM THE PATCH AREA
2158  FF39              ; IS EMPTY
2159  FF39              ;---------------------------------------------
2160  FF39              ;
2161  FF39              ;---------------------------------------------
2162  FF39              ; TABLE: SYSTEM VECTORS
2163  FF39              ;----------------------------------------------------
2164  FF39              ;
2165  FF39              ;DEFAULT TABLE FOR USER COMMAND
2166  FF39              ;
2167  FF39                     *= $FFE6
2168  FFE6              ;--------------------------------------------------
2169  FFE6  15 FB              .WORD FORMT    ;FORMAT A DISK (1541 USES FORMAT)
2170  FFE8  DD F9              .WORD TRNOFF   ;SWITCH OFF DRIVE MOTOR
2171  FFEA  5F CD       UBLOCK .WORD UBLKRD    ;U1 COMMAND
2172  FFEC  97 CD              .WORD UBLKWT   ;U2 COMMAND
2173  FFEE              ;
2174  FFEE              ; DEFAULT TABLE FOR USER COMMAND
2175  FFEE  00 05              .WORD $0500    ;LINKS TO BUFFER #2
2176  FFF0  03 05              .WORD $0503
2177  FFF2  06 05              .WORD $0506
2178  FFF4  09 05              .WORD $0509
2179  FFF6  0C 05              .WORD $050C
2180  FFF8  0F 05              .WORD $050F
2181  FFFA              ; STANDARD 6502 VECTORS
2182  FFFA                     *= $FFFA
2183  FFFA  36 FF              .WORD NMI      ;NMI ADDRESS
2184  FFFC  C5 EA              .WORD DSKINT   ;RESET ADDRESS
2185  FFFE  B6 FE              .WORD SYSIRQ   ;IRQ ADDRESS
2186  0000              ;.END
2186  0000              ;
2187  0000                                    ;.LIB CHKSUM
2188  0000                     .END


ERRORS = 0000


SYMBOL TABLE

SYMBOL VALUE
 AC10     F279    AC20     F27B    AC30     F284    AC40     F294
 ACC200   CEE6    ACCUM    008F    ACCX2    CEE5    ACCX4    CEE2
 ACLSTP   0060    ACLTIM   0048    ACR1     180B    ACR2     1C0B
 ACTJOB   001B    AD10     C325    ADD100   CEF0    ADDFIL   D6E4
 ADDNR    E304    ADDR1    E33B    ADDREL   E31C    ADDRES   CEED
 ADDRTS   DF65    ADDSUM   E879    ADDT12   DF5C    ADRSED   007B
 AF       005F    AF08     D715    AF10     D726    AF15     D730
 AF20     D73D    AF25     D74D    AF30     D76F    AF50     D790
```

```
SYMBOL TABLE

SYMBOL VALUE
 AH10     CCAB    AH20     CCCA    AH30     CCD0    AH35     CCD7
 AH40     CCE4    ALLDRS   C320    AN05     E318    AN10     E31B
 AP30     DA42    APMODE   0002    APPEND   DA2A    AR10     E355
 AR20     E363    AR25     E368    AR30     E372    AR35     E38F
 AR40     E39D    AR45     E3B6    AR50     E3C8    AR55     E3D4
 AR60     E3F9    AR65     E418    AR70     E444    AS       005E
 ASCHEX   CCA1    ATN      0080    ATN10    E8AA    ATN20    E8B4
 ATN30    E8BD    ATN40    E932    ATN50    E93A    ATN60    E94C
 ATN70    E95D    ATNA     0001    ATNIRQ   E881    ATNMOD   007D
 ATNPND   007C    ATNSRV   E889    AUTO1    C65F    AUTO2    C669
 AUTOFG   0068    AUTOI    C63D    AVCK     F26E    B02      DEB9
 B0TOB0   DEA5    B2X10    F166    BA10     CD06    BA15     CD19
 BA20     CD1A    BA30     CD2C    BA40     CD31    BADCMD   0031
 BADFN    0033    BADID    F469    BADSYN   0030    BADTS    0066
 BAM      02A1    BAM2A    F15D    BAM2X    F167    BAMSIZ   FED5
 BCD2     E6B4    BCDDEC   E6AB    BCJMP    CC63    BCTAB    CC5D
 BE05     CDAB    BE10     CDBA    BFCNT    0005    BGTAB    F7CD
 BH10     E2DC    BHERE    E2D0    BHERE2   E2D3    BID      0038
 BIGFIL   0052    BING07   F808    BING20   F827    BING35   F7BD
 BINGCR   F7DD    BITCNT   0037    BKOTST   CDF2    BLINDX   0006
 BLK10    CC26    BLK30    CC2B    BLK40    CC30    BLK50    CC38
 BLK60    CC42    BLKALC   CD03    BLKEXC   CDA3    BLKFRE   CCF5
 BLKNB    C7AC    BLKNB1   C7B0    BLKPAR   CC6F    BLKPTR   CDBD
 BLKRD    CD56    BLKRD2   CD36    BLKRD3   CD42    BLKTST   CDF5
 BLKWT    CD73    BLOCK    CC1B    BMASK    F037    BMP      F3CA
 BMPNT    006D    BOOT     E790    BOOT4    E7D6    BP05     CC7C
 BP10     CC8B    BP20     CC92    BREAK    E202    BT05     CE08
 BT15     CDE0    BT20     CDE5    BTAB     0052    BUF0     00A7
 BUF1     00AE    BUFF0    0300    BUFF1    0400    BUFF2    0500
 BUFIND   FF2F    BUFPNT   0030    BUFS     0300    BUFTAB   0099
 BUFTST   CDD2    BUFUSE   024F    BUMP     00C0    BUMPC    0040
 BW10     CD81    BW20     CD8C    BYTCNT   0036    BYTE     003C
 CB       00A3    CB10     DECC    CBMV2    0073    CBPTR    000A
 CC10     C594    CC15     C59A    CC20     C5A6    CCHKSM   C000
 CDIRTY   E115    CDRIVE   003E    CHAR     0275    CHKB10   F63A
 CHKBLK   F637    CHKIN    CACC    CHKIO    CAE7    CHKSUM   003A
 CHNDAT   023E    CHNRDY   00F2    CJUMPH   FEF0    CJUMPL   FEE4
 CK10     CAD6    CK20     CAE6    CK25     CAEA    CK30     CAF4
 CKM1     DA11    CKM2     DA1C    CKT1     DA1E    CKT2     DA29
 CKTM     DA09    CLB1     F059    CLD2     D334    CLDCHN   D313
 CLEAR    FE5D    CLER10   FE75    CLNBAM   F11F    CLOSE    DAC0
 CLR10    E104    CLRB2    C1C1    CLRBAM   F053    CLRBUF   DEC1
 CLRC1    D30B    CLRCB    C1BD    CLRCHN   D307    CLREC    E0F3
 CLRF10   DDA3    CLRFLG   DD9D    CLS05    DAD1    CLS10    DAD4
 CLS15    DAE9    CLS20    DAF0    CLS25    DAFF    CLSALL   DAEC
 CLSC28   DB0C    CLSC30   DB26    CLSC31   DB29    CLSCHN   DB02
 CLSD     D317    CLSD4    DC06    CLSD5    DC21    CLSD6    DC29
 CLSDIR   DBA5    CLSR1    DB5F    CLSREL   DB2C    CLSW10   DB76
 CLSW15   DB88    CLSW20   DB8C    CLSWRT   DB62    CMD      024D
 CMDBUF   0200    CMDCHN   0004    CMDER2   E645    CMDER3   E648
 CMDERR   C1C8    CMDLEN   0029    CMDNUM   022A    CMDRST   C2DC
 CMDSA    000F    CMDSET   C2B3    CMDSIZ   0274    CMDTBL   FED8
 CMDWAT   0255    CMPCHK   C589    CMPR10   FD88    CMPR15   FD8F
 CMPR20   FDA7    CNT      0620    CNT10    FC09    CNT20    FC1C
 CNTINT   F2A7    CNTST    F2A7    CNVBIN   F4E5    CODE     FB15
```

```
SYMBOL TABLE

SYMBOL VALUE
 COMP     FD7B    COMPAR   C4D8    CONHDR   F982    CONT     0098
 CONT10   F30E    CONT20   F341    CONT30   F31B    CONT35   F326
 CONT40   F337    CONT50   F33E    COP01    C982    COP05    C987
 COP10    C9A1    COPY     C952    COUNT    FC04    CP02     C4E6
 CP05     C4E7    CP10     C4EC    CP20     C4FE    CP30     C50A
 CP32     C51B    CP33     C51D    CP34     C52B    CP40     C535
 CP42     C55C    CP50     C589    CR       000D    CR20     EB1C
 CR30     EB1E    CRTDAT   FCD7    CS07     C2CA    CS08     C2CB
 CS10     C2FD    CSECT    004C    CSERR    F46C    CTRACK   0046
 CURBLK   DE3B    CY       C9A7    CY10     C9B9    CY10A    C9CE
 CY15     C9D5    CY20     C9D8    CY30     C9EA    CYEXT    CA53
 DATA     0085    DATA2    1C01    DAV      0040    DBID     0047
 DBL05    CF57    DBL08    CF5D    DBL10    CF66    DBL15    CF6C
 DBL20    CF6F    DBL30    CF76    DBLBUF   CF1E    DBS10    CF8B
 DBSET    CF7B    DBSYNC   FD2F    DCDE     E8D8    DCDE20   E8EC
 DCDE30   E8F8    DCDE40   E8FD    DCDE50   E901    DCDE60   E90B
 DCDE70   E919    DCDE80   E932    DDRA1    1803    DDRA2    1C03
 DDRB1    1802    DDRB2    1C02    DECTAB   CCF2    DEL1     C8AD
 DEL10    C894    DEL2     C894    DELDIR   C8B6    DELFIL   C87D
 DELIND   0292    DELSEC   0291    DERR     F243    DIAGOK   EB54
 DIND     0266    DIR1     ED38    DIR10    ED5B    DIR3     ED71
 DIRBUF   0094    DIRERR   0071    DIRLEN   0018    DIRLST   0254
 DIRSEC   0290    DIRTRK   FED4    DIRTYP   0007    DIV100   CE77
 DIV120   CE71    DIV150   CE87    DIV200   CE89    DIV254   CE6E
 DIV300   CEA3    DIV400   CEB0    DIV500   CEBF    DIV600   CED6
 DIV700   CED8    DKIT10   EACF    DOIT     D590    DOIT2    D593
 DOITT    F4C1    DOJOB    D58C    DONE     F466    DOREAD   D586
 DOREC    D6A6    DOREC1   D6AB    DOREC2   D6B9    DOREC3   D6C4
 DOSTEP   FA7C    DOSVER   0002    DOWRIT   D58A    DRDBYT   D4F6
 DRIVE    003D    DRT      D466    DRTRD    D460    DRTWRT   D464
 DRVCNT   028C    DRVFLG   028D    DRVNUM   007F    DRVST    0020
 DRVTRK   0022    DS08     FC2E    DS10     FC38    DS14     FC3F
 DS15     FC51    DS17     FC63    DS20     FC67    DS22     FC6F
 DS30     FC72    DS32     FC7E    DSEC     0260    DSKCNT   1C00
 DSKCPY   C8F0    DSKFUL   0072    DSKID    0012    DSKIN1   EBD6
 DSKIN2   EBDE    DSKINT   EAC5    DSKNAM   FED7    DSKVER   0101
 DSTRT    F558    DTRCK    0626    DTYBAM   EFD6    DUPLCT   C8C1
 DX0000   C90C    DX0005   C90F    DX0010   C923    DX0020   C928
 DYFILE   0040    E10      E718    E20      E722    E30      E727
 E40      E735    E45      E739    E50      E73D    E55      E742
 E60      E754    E70      E763    E90      E74D    EA10     E76D
 EA20     E77E    EADV1    E767    EADV2    E775    ECHKSM   FF35
 END      F9EA    END001   F9F3    END002   F9FF    END003   FA0F
 END10    FA27    END20    FA32    END30    FA48    END30X   FA50
 END33    FB0C    END33X   FA24    END40    FA19    ENDCMD   C194
 ENDRD    D7EB    ENDSAV   C199    ENTFND   0253    ENTIND   00DD
 ENTSEC   00D8    EOI      0008    EOIFLG   00F8    EOIOUT   0080
 EOISND   0008    EPTR     0299    ER0      FF1C    ER00     FF1B
 ER1      FF1D    ER2      FF1E    ER3      FF1F    ERLED    026D
 ERMOVE   E706    ERMSG2   E6E3    ERR      F5A1    ERR1     E625
 ERR10    E698    ERR2     E627    ERR3     E62D    ERR4     E644
 ERRBUF   02D5    ERRCHN   0005    ERRMSG   E6C7    ERROFF   C123
 ERRON    C12C    ERROR    E60A    ERRR     F9B7    ERRR10   F9C3
 ERRSA    0010    ERRTAB   E4FC    ERRTS0   E6C1    ERWORD   026C
 ESEEK    F45E    ETEND    E60A    EX       F3BC    EX2      F3BE
```

```
SYMBOL TABLE

SYMBOL VALUE
 EX3      F3C7     EXE      F3B5     EXEC     00E0     EXECD    0060
 F000     FB91     F001     FB94     F005     FBAA     F006     FBB2
 F007     FBB5     F009     FBCB     F013     FBE5     F014     FBEC
 F1CNT    0277     F1PTR    00D3     F2CNT    0278     F2PTR    0279
 FB1      D2BC     FB2      D2C8     FB3      D2D8     FBAM10   F052
 FBG10    FE93     FBG15    FEB3     FBTOG    FE7F     FBUFS    0300
 FF10     C4AA     FF15     C492     FF25     C4BA     FF30     C4C9
 FF40     C4D7     FFRE     C48B     FFST     C49D     FILCNT   0295
 FILDRV   00E2     FILNOP   0061     FILOPN   0060     FILSEC   0285
 FILTBL   027A     FILTRK   0280     FILTYP   00EC     FIN      F36B
 FIXBAM   F03F     FL05     C6AD     FL10     C6C8     FLEXST   0063
 FLNTFD   0062     FM2030   0042     FM4040   0041     FMT105   C8E0
 FMT110   C8EF     FMTE10   FE2A     FMTEND   FDE5     FMTERR   FE22
 FMTVAR   0620     FND10    D383     FND2     F243     FND30    D391
 FNDBUF   D2BA     FNDC20   D0F3     FNDC25   D0F9     FNDC30   D106
 FNDFIL   C4B5     FNDL10   E1B7     FNDL20   E1C8     FNDL30   E1C4
 FNDLMT   C6A6     FNDLNX   D37F     FNDLST   E1B2     FNDN0    F1E3
 FNDN1    F1E8     FNDN2    F1EB     FNDNXT   F1C1     FNDRCH   D0EB
 FNDREL   CE0E     FNDSEC   F234     FNDW10   D119     FNDW13   D10F
 FNDW15   D121     FNDW20   D123     FNDWCH   D107     FORMAT   C8C6
 FORMT    FB15     FOUND    028F     FRE10    EFD5     FRE25    D259
 FREB1    D2F9     FREB2    D303     FREBUF   D2F3     FRECHN   D227
 FRECO    D22E     FREEC0   C001     FREIAC   D2DA     FREICH   D4DA
 FREMSG   C817     FRERD    D22E     FRERTS   EFD5     FRETS    EFAD
 FRETS2   EFB0     FREUS2   F021     FREUS3   F023     FREUSE   F01D
 FREWRT   D22E     FRI10    D2E9     FRI20    D2D9     FS10     C3B0
 FS15     C3B8     FS1SET   C398     FTNUM    0051     FWAIT    FB87
 FWAIT2   FB8C     GA1      DF9B     GA2      DFA0     GA3      DFB0
 GAFLGS   DF9E     GAP1     000A     GAP2     0004     GBERR    D20F
 GBF1     D2A3     GBF2     D2B6     GBYTE    D39B     GCBYTE   CA39
 GCRB10   F95A     GCRB20   F979     GCRBIN   F92E     GCRERR   0035
 GCRFLG   0050     GCRHI    F8EE     GCRLO    F90E     GCRPNT   0034
 GD1      EDCC     GE10     D433     GE15     D43A     GE20     D443
 GE30     D445     GET      D3AA     GET0     D3CE     GET00    D3B4
 GET1     D3D3     GET2     D3D7     GET3     D403     GET4GB   F834
 GET6     D409     GETACT   DF93     GETB1    D151     GETB2    D14D
 GETBF    D1F8     GETBUF   D28E     GETBYT   D137     GETD3    EDBB
 GETDIR   EDB5     GETERC   D414     GETHDR   DE3E     GETINA   DFB7
 GETLNK   DE0C     GETNAM   C6CE     GETPNT   D4E8     GETPRE   D12F
 GETR2    D1E3     GETR3    D206     GETR4    D226     GETR5    D217
 GETR52   D1F3     GETR55   D1F5     GETRCH   D1E2     GETSEC   F248
 GETSIM   CD3C     GETWCH   D1DF     GI10     DFBF     GIB20    CA52
 GIBYTE   CA35     GN05     C6F7     GN050    C6FC     GN051    C70E
 GN10     C71B     GN12     C728     GN14     C73C     GN15     C74A
 GN20     C76B     GN22     C773     GN30     C783     GN35     C793
 GN37     C798     GN40     C7A7     GN45     C7AB     GNSUB    C6DE
 GOTU     F38A     GOTU10   F390     GP1      D4EB     GS10     F25B
 GS20     F26B     GS30     F26D     GSSPNT   DF45     GTAB     0056
 GTABYE   E876     GTABYT   E867     HBID     0039     HDRPNT   0032
 HDRS     0006     HEADER   0016     HED2TS   D552     HEDOFF   D676
 HEX0     E69F     HEX5     E6AA     HEXDEC   E69B     HINIB    003B
 HOF1     D67C     HOF2     D688     HOF3     D692     IBOP     DF25
 IBRD     DF1B     IBWT     DF21     ICMD     0096     ID20     D00B
 IDL01    EC55     IDL1     EC4D     IDL10    ECCF     IDL11    ECD9
 IDL12    ECE6     IDL2     EC60     IDL3     EC79     IDL4     EC7F
```

```
SYMBOL TABLE

SYMBOL VALUE
 IDL5     EC89    IDL6     ECA6    IDL7     ECAA    IDL8     ECB7
 IDL9     ECBB    IDLE     EC3D    IEEED    1801    IER1     180E
 IER2     1C0E    IFR1     180D    IFR2     1C0D    ILERR    EA85
 IMAGE    028B    INAC10   FA5C    INAC20   FA6A    INACT    FA53
 INCPNT   D1C6    INCPTR   D1C6    INDEX    0294    INITDR   D042
 INITP    DCB6    INTDRV   D005    INTT1    EBAF    INTTAB   EBAB
 INTTS    F1F7    IP       0075    IRQ10    FEC5    IRQ20    FECE
 IRSA     0011    ISR04    EA44    IT20     D024    IT30     D02C
 ITERR    EA7C    ITRIAL   D00E    ITS1     F206    ITS2     F219
 ITS3     F22D    IWSA     0012    JMPC     0050    JOB      0045
 JOBN     003F    JOBNUM   00F9    JOBRTN   0298    JOBS     0000
 JUMPC    00D0    KILL     FE4F    KILLP    F2A6    L213     FB43
 L214     FB4E    L40      CFBF    L41      CFC9    L42      CFD8
 L45      CFED    L46      CFCE    L460     F480    L465     F4AF
 L470     F4D1    L480     F488    L490     F4DB    L50      CFE8
 LBUSED   0257    LCC      F2FE    LD01     DA62    LD02     DA6D
 LD03     DA86    LD05     DA90    LD10     DA9E    LD20     DAA7
 LDCMD    000C    LE77F    E77F    LE7BB    E7BB    LED0     0008
 LED1     0000    LEDMSK   FF19    LEDOUT   1C02    LEDPRT   1C00
 LEDS0    C110    LEDS1    C113    LEDSON   C118    LIMIT    0276
 LINDX    0082    LINTAB   022B    LINUSE   0256    LISNER   0001
 LISTEN   E960    LK05     C452    LK10     C45C    LK15     C462
 LK20     C470    LK25     C475    LK26     C47E    LK30     C485
 LOADIR   DA55    LONGLN   0032    LOOKUP   C44F    LRF      0080
 LRUEXT   CF1D    LRUILP   CEFC    LRUINT   CEFA    LRULP1   CF0D
 LRUTBL   00FA    LRUUPD   CF09    LSN10    E969    LSN15    E97A
 LSN20    E982    LSN21    E988    LSN25    E99B    LSN28    E9AC
 LSN29    E9C3    LSN30    E9CD    LSN40    E9E9    LSNACT   0079
 LSNADR   0077    LSNERR   E688    LSTBUF   0293    LSTCHR   0244
 LSTDRV   028E    LSTJOB   025B    LSTSEC   024E    LWPT     001E
 LXINT    000F    M10      CB50    M30      CB45    MAK10    FC8D
 MAPOFF   FED6    MAPOUT   EF42    MASK1    00F8    MASK2    0007
 MASK2X   00C0    MASK3    003E    MASK4    0001    MASK4X   00F0
 MASK5    000F    MASK5X   0080    MASK6    007C    MASK7    0003
 MASK7X   00E0    MASK8    001F    MAX1     F29C    MAXSA    0012
 MAXSEC   F299    MAXTRK   FF26    MDIRTY   0251    MDMODE   0003
 MEM      CAF8    MEMERR   CB4B    MEMEX    CB1D    MEMRD    CB20
 MEMWRT   CB50    MH10     D69A    MINSTP   0064    MISTYP   0064
 MO10     EF49    MODE     0297    MODLST   FF01    MOVB1    EDA9
 MOVBUF   EDA7    MOVHED   D693    MOVO10   FE46    MOVOVR   FE44
 MOVUP    FE34    MRK1     EE52    MRK2     EE3C    MSG1     C80B
 MSGFRE   C806    MSGLEN   000C    MUL100   CE50    MUL200   CE57
 MUL25    CE41    MUL400   CE6D    MUL50    CE4C    MULPLY   CE2C
 MXCHNS   0006    MXFILS   0005    MYPA     0097    N101     EE67
 N108     EE94    N110     EEA4    NAMBUF   02B1    NB10     E2E9
 NB20     E2F1    NB30     E303    NBCMDS   0006    NBKH     00BB
 NBKL     00B5    NBSIZ    001B    NBTEMP   0272    NCMDS    000C
 ND10     C7DC    ND15     C7E5    ND20     C7ED    NDAC     0004
 NDBH     02FC    NDBL     02FA    NEW      EE5B    NEWDIR   C7B7
 NEWMAP   EF05    NEWMPV   EF05    NEWSS    E44E    NEXTS    004D
 NFCALC   D075    NM10     EF15    NM20     EF27    NM30     EF31
 NMI      FF36    NMIFLG   0067    NMODES   0004    NOBLK    0065
 NOCFIL   0039    NOCHNL   0070    NODRIV   0074    NODRV    00FF
 NOFILE   0034    NOLATN   E998    NOREC    0050    NOTATN   EA79
 NOTFND   0006    NOTLK    EA16    NOTRDY   0000    NOTYET   D5C4
```

```
SYMBOL TABLE

SYMBOL VALUE
 NR        00C1     NRBU20   E07B     NRBU50   E05D     NRBU70   E06B
 NRBUF     E03C     NRFD     0002     NS20     E4AC     NS40     E4D1
 NS50      E4DE     NSSL     0006     NSSP     0078     NTYPES   0005
 NULBUF    E2E2     NULLNK   DE19     NUM      0621     NUMF1    D07D
 NUMF2     D083     NUMFRE   EF9B     NUMJOB   0006     NUMSEC   FF20
 NUMSYN    0005     NXDB1    D4BB     NXDRBK   D48D     NXOUT    E009
 NXT1      F17B     NXT2     F1AD     NXTB1    D45F     NXTBF    004E
 NXTBUF    D44D     NXTDS    F17B     NXTERR   F1A8     NXTJOB   0041
 NXTPNT    004F     NXTR15   DFE4     NXTR20   DFF6     NXTR30   E034
 NXTR35    E035     NXTR40   E01D     NXTR45   E018     NXTR50   E02A
 NXTREC    DFD0     NXTRK    0042     NXTST    0062     NXTTS    F16C
 NZONES    FF25     OB05     CBA0     OB10     CBA5     OB15     CBB8
 OB30      CBF1     OFFSET   FF2A     OK       D5C2     OKERR    E6BC
 ONEDRV    C312     OP02     D7CF     OP021    D7F3     OP04     D7FF
 OP041     D815     OP0415   D81C     OP042    D82B     OP049    D834
 OP05      D837     OP10     D83C     OP100    D94A     OP110    D95C
 OP115     D965     OP120    D96A     OP125    D990     OP130    D9C3
 OP20      D840     OP40     D876     OP45     D891     OP50     D8A7
 OP60      D8B1     OP75     D8C6     OP77     D8CD     OP80     D8D9
 OP81      D8E1     OP815    D8F0     OP82     D8F5     OP90     D940
 OP95      D945     OPEN     D7B4     OPF1     DA06     OPFIN    D9EF
 OPIR10    CA31     OPIRFL   C9FA     OPNBLK   CB84     OPNIRD   D475
 OPNIWR    D486     OPNRCH   DC46     OPNTYP   D477     OPNWCH   DCDA
 OPREAD    D9A0     OPTSCH   C3CA     OPWRIT   D9E3     OR10     DC65
 OR20      DC81     OR30     DCA9     ORGSA    0084     OROW     DC98
 OS10      C3D5     OS15     C3E8     OS30     C3EF     OS35     C400
 OS40      C43C     OS45     C41B     OS50     C420     OS60     C434
 OS70      C439     OVRBUF   0100     OVRFLO   0020     OW10     DCFD
 OW20      DD16     P10      E2AA     P2       E289     P30      E291
 P75       E2BF     P80      E2C2     PA1      1801     PARSE    C268
 PARSXQ    C146     PATCH    FF39     PATFLG   028A     PATTYP   00E7
 PB        1800     PBYTE    CFAF     PCMD     0009     PCR1     180C
 PCR2      1C0C     PD10     EAA3     PD11     EAB4     PD20     EAA4
 PD21      EAB5     PE20     EA96     PE30     EA97     PE40     EAB3
 PERR      EA93     PERR2    EB51     PEZRO    EA90     PHASE    02FE
 PI1       DFCD     PIBYTE   CF9B     POSBUF   E29C     POSITN   E275
 PR10      C26B     PR20     C280     PR25     C283     PR28     C299
 PR30      C29E     PR35     C2A0     PR40     C2B1     PRGDRV   026E
 PRGSEC    026F     PRGTRK   007E     PRGTYP   0002     PRSCLN   C1E5
 PS05      C160     PS10     C16A     PS20     C17A     PS30     C184
 PU10      EADE     PU20     EAE4     PU30     EAE9     PUPS1    C932
 PUT       CFB7     PUT4BG   F71E     PUTB1    CFFD     PUTBAM   F0F3
 PUTBYT    CFF1     PUTINA   DFC2     PUTSS    DD8D     QUE      F347
 QUE05     F354     QUE10    F34F     QUE20    F36E     QUIT     D635
 QUIT2     D63F     R0       0086     R1       0087     R2       0088
 R20       E219     R3       0089     R30      E228     R35      E253
 R4        008A     R40      E253     R50      E265     R60      E272
 RA10      EB24     RA30     EB34     RA40     EB36     RAMEND   0300
 RAMTST    EB22     RBM10    F140     RBM20    F158     RD0      D15D
 RD01      D164     RD05     E15E     RD1      D16A     RD10     E127
 RD15      E138     RD20     E13B     RD25     E13D     RD3      D191
 RD30      E14D     RD4      D192     RD40     E153     RDAB     DE57
 RDBUF     D0C3     RDBYT    D156     RDIN     DE65     RDLNK    DE95
 RDMAX     0006     RDMODE   0000     RDREL    E120     RDS5     DE75
 RDSS      DE73     RDYLST   0001     RDYTLK   0088     READ     0080
```

```
SYMBOL TABLE

SYMBOL VALUE
 READ01   F51F    READ11   F522    READ20   F52F    READ28   F549
 READ40   F553    REC      0258    REC0     D5F4    REC01    D5E3
 REC1     D600    REC3     D625    REC5     D631    REC7     D644
 REC8     D651    REC9     D65C    REC95    D66D    RECH     00BB
 RECL     00B5    RECORD   E207    RECOV    D5C6    RECOVF   0051
 RECPTR   00D4    REDBAM   F12D    REED     F518    REL1     D26B
 REL10    D253    REL15    D24D    REL2     D27C    REL3     D28D
 RELBUF   D25A    RELINX   D249    RELP05   E096    RELP06   E094
 RELP07   E09E    RELP10   E0A3    RELP20   E0AA    RELPTR   00D7
 RELPUT   E07C    RELTYP   0004    REMDR    0627    RENAME   CA88
 RESULT   008B    REVCNT   006A    RLINDX   0271    RM10     EAFB
 RN10     CA97    RNDEOI   0081    RNDGET   D3DE    RNDRDY   0089
 RNGET1   D3EC    RNGET2   D3EE    RNGET3   D3FF    RNGET4   D3F0
 ROM      C000    RS       00C7    RSTEPS   0061    RT10     EB07
 RT20     EB09    SA       0083    SA05     C370    SA10     C383
 SA20     C388    SAVPNT   002E    SAVSP    0049    SB10     EF55
 SB20     EF72    SBM10    F070    SBM30    F08C    SC15     C835
 SC17     C855    SC20     C86B    SC25     C86D    SC30     C872
 SCAL1    DF4C    SCFLG    DD95    SCHTBL   C440    SCR1     DDFC
 SCRBAM   EF4D    SCREN1   C1AD    SCREND   C1A3    SCRTCH   C823
 SCRUB    DDF1    SD20     C34C    SD22     C34D    SD24     C34F
 SD40     C352    SD50     C361    SDIRTY   E105    SE10     E1D8
 SE20     E1DC    SE30     E1EF    SEAK     F3FF    SEARCH   C617
 SECINC   0069    SECSEK   00B8    SECSS    025A    SECT     0628
 SECTOR   0081    SECTR    0043    SECTSK   F718    SEEK     00B0
 SEEK10   F409    SEEK15   F416    SEEK20   F455    SEEK30   F42A
 SEQGET   D400    SEQTYP   0001    SET00    DE2B    SETANY   C368
 SETBAM   F05F    SETBPT   EF88    SETDIR   D4EB    SETDRN   D1D3
 SETDRV   C33C    SETERR   EC35    SETFLG   DD97    SETH     D6D3
 SETHDR   D6D0    SETJ10   F3F9    SETJB    F3E1    SETJB1   F3E3
 SETJOB   D50E    SETL01   E17E    SETL05   E19D    SETL10   E1A4
 SETL40   E1AC    SETLDS   C100    SETLE    FA9C    SETLJB   D506
 SETLNK   DDFD    SETLST   E16E    SETPNT   D4C8    SETSSP   DEE9
 SHORT    FA89    SIMPRS   C1D1    SJ10     DE7F    SJ20     DE8B
 SJ30     DE92    SJB1     D57A    SJB2     D535    SJB3     D538
 SJB4     D53F    SKIP2    002C    SP10     C1E2    SR1      180A
 SR10     C5C4    SR15     C5CA    SR2      1C0A    SR20     C5D7
 SR30     C5FB    SR40     C629    SR50     C62F    SRCH     F55E
 SRCH20   F586    SRCH25   F58B    SRCH30   F59C    SRCHST   C5AC
 SRRE     C604    SS       00CD    SSA10    FAE2    SSACL    FAC9
 SSCALC   DF51    SSDEC    FAF3    SSDIR    DEDC    SSEND    E1CB
 SSIND    00D6    SSIOFF   0010    SSNUM    00D5    SSP10    DF0B
 SSP20    DF12    SSPOS    DEF8    SSRUN    FAE5    SSSET    DED2
 SSTST    DF66    ST10     DF77    ST20     DF7B    ST30     DF8B
 ST40     DF8F    STAB     0024    STDIR    ECEC    STEPS    004A
 STL05    D33E    STL10    D348    STL20    D355    STL30    D35E
 STL40    D363    STL50    D373    STL60    D37A    STLBUF   D339
 STP      FAB7    STPIN    FAB1    STPOUT   FA80    STR1     D0B7
 STRDBL   D0AF    STRRD    D09B    STRSIZ   024B    STRTIT   D0C9
 STRUCT   FEF4    SWAP     F0A9    SWAP1    F10C    SWAP2    F11E
 SWAP3    F0CD    SWAP4    F0ED    SYC10    FE08    SYNC     F5A4
 SYNC10   F5AB    SYNCLR   FDF2    SYSIRQ   FEB6    SYSTS    0067
 T0       006F    T0V1     C3C7    T1       0070    T1HC1    1805
 T1HC2    1C05    T1HL1    1807    T1HL2    1C07    T1LC1    1804
 T1LC2    1C04    T1LL1    1806    T1LL2    1C06    T2       0071
```

```
SYMBOL TABLE

SYMBOL VALUE
 T2HC1    1809    T2LC1    1808    T2LH2    1C09    T2LL2    1C08
 T3       0072    T4       0073    TAGCMD   C1EE    TALK     EA0A
 TALK1    EA10    TALKER   0080    TBAM     029D    TC25     C1F3
 TC30     C1F8    TC35     C200    TC40     C20A    TC50     C228
 TC60     C245    TC70     C24C    TC75     C254    TC80     C260
 TEMP     006F    TEMPSA   024C    TGLBUF   CF8C    TIM      003A
 TIMER1   1805    TJ10     D5BA    TLERR    E68E    TLK05    EA16
 TLK10    EA17    TLK20    EA2B    TLK25    E9FF    TLK30    EA53
 TLK35    EA5A    TLK40    EA70    TLKACT   007A    TLKADR   0078
 TLKERR   E680    TLKRTN   EA61    TMP      004B    TN10     C681
 TN20     C687    TOBIG    0004    TOFF     029A    TOGDRV   C38F
 TOLONG   0002    TOMANY   0003    TOP      F30C    TOPP     FB5A
 TOPRD    0045    TOPWRT   0045    TOSMAL   0005    TP1LST   FF0F
 TP2LST   FF14    TPLST    FF05    TR       0010    TR10     C697
 TR20     C6A5    TRACC    0040    TRACK    0080    TRAL     0624
 TRCMBF   C688    TRKNUM   FF26    TRKSS    0259    TRNAME   C66E
 TRNOFF   F9DD    TRYS     0623    TSCHK    D55F    TSER1    D54D
 TSERR    D54A    TST05    FDB6    TST0V1   C3BD    TST10    FDC6
 TSTC20   DDB9    TSTC30   DDC2    TSTC40   DDCA    TSTCHN   DDB7
 TSTDAT   FDB1    TSTFLG   DDA6    TSTJOB   D5A6    TSTRDJ   F4CC
 TSTRTS   DDC9    TSTWRT   DDAB    TURNON   F9CC    TYPE     024A
 TYPFIL   D125    TYPFLG   0296    TYPLST   FF0A    TYPMSK   0007
 UBAM     029B    UBLKRD   CD5F    UBLKWT   CD97    UBLOCK   FFEA
 UNLSN    003F    UNTLK    005F    US10     CB6C    USE10    F008
 USE20    F00B    USEDTS   EFE1    USER     CB5C    USERTS   F01C
 USREXC   CB72    USRINT   CB63    USRJMP   006B    USRTYP   0003
 UTLD00   E7F3    UTLD10   E806    UTLD20   E828    UTLD30   E830
 UTLD35   E845    UTLD50   E85A    UTLODR   E7D1    VAL      0000
 VALDAT   EDD2    VD10     EDEA    VD15     EE01    VD17     EE19
 VD20     EE22    VD25     EE27    VERDIR   EDD2    VERERR   0007
 VERNUM   FF24    VMKBAM   EE33    VNERR    D572    VNMI     0065
 VRF10    F6E6    VRF15    F6F3    VRF20    F713    VRF30    F701
 VRFY     F6DF    WATJOB   D599    WBAM     02F9    WFREE    EFAA
 WGP2     FD58    WLINDX   0270    WORK     0044    WPSW     001C
 WR10     E0B2    WR20     E0B7    WR30     E0BC    WR40     E0C8
 WR45     E0D6    WR50     E0D9    WR51     E0E1    WR60     E0E2
 WRIGHT   F5BC    WRITE    0090    WRT0     D1A3    WRT05    F5C3
 WRT10    F5D4    WRT20    F5DC    WRT30    F601    WRT40    F60D
 WRTAB    DE50    WRTBUF   D0C7    WRTBYT   D19D    WRTC1    D0E8
 WRTMAX   000C    WRTMIN   0009    WRTN10   FE18    WRTNUM   FE12
 WRTOUT   DE5E    WRTREL   E0AB    WRTS10   FD07    WRTS20   FD11
 WRTS30   FD20    WRTS40   FD3A    WRTS50   FD48    WRTSNC   F5F9
 WRTSS    DE6C    WRTSYN   FD00    WSECT    F471    WTMODE   0001
 WTOB14   F672    WTOB50   F691    WTOB52   F6BC    WTOB53   F69D
 WTOB55   F6C6    WTOB57   F6D3    WTOBIN   F640    WUSED    EFDE
 WVERFY   00A0    XX05     F850    XX06     F8A8    ZERRES   CED9
 ZP2      0065    ZPEND    0103

END OF ASSEMBLY
```
