```
000077  SEL3            .BYTE       000,0D4,0D2,0D3,0D3
000078
000079  FOUND           .BLOCK      16.                     ;'SECTOR FOUND' TABLE.

; ##########################################################################################################
; #   END OF FILE:  FMTDATA.TEXT
; #   LINES      :  79
; #   CHARACTERS :  4060
; #   Formatter  :  Assembly Language Reformatter 1.0.2 (06 January 1998)
; #   Author     :  David T. Craig -- 71533.606@compuserve.com -- Santa Fe, New Mexico USA
; ##########################################################################################################
```

```
; ##############################################################################################################
; #   PROJECT  :  Apple /// SOS DISK /// Formatter Device Driver 1.30 (6502 Assembly Source Code)
; #   FILE NAME:  FMTREAD.TEXT
; ##############################################################################################################

000001                  .TITLE       "16-SECTOR READ"
000002                  .PAGE
000003  ;************************
000004  ;                       *
000005  ;     READ SUBROUTINE   *
000006  ;   (16-SECTOR FORMAT)   *
000007  ;                       *
000008  ;************************
000009  ;    READS 6-BIT NIBLS   *
000010  ;     (00ABCDEF) INTO    *
000011  ;     NBUF1 AND NBUF2    *
000012  ;    CONVERTING 7-BIT    *
000013  ;     NIBLS TO 6-BIT     *
000014  ;    VIA 'DNIBL' TABLE   *
000015  ;                       *
000016  ;  FIRST READS NBUF2     *
000017  ;          HIGH TO LOW,  *
000018  ;  THEN READS NBUF1      *
000019  ;          LOW TO HIGH.  *
000020  ;                       *
000021  ;   ---- ON ENTRY ----   *
000022  ;                       *
000023  ;  X-REG: SLOTNUM       *
000024  ;         TIMES $10.    *
000025  ;                       *
000026  ;  READ MODE (Q6L, Q7L)  *
000027  ;                       *
000028  ;   ---- ON EXIT -----   *
000029  ;                       *
000030  ;  CARRY SET IF ERROR.   *
000031  ;                       *
000032  ;  IF NO ERROR:         *
000033  ;     A-REG HOLDS $AA.  *
000034  ;     X-REG UNCHANGED.  *
000035  ;     Y-REG HOLDS $00.  *
000036  ;     CARRY CLEAR.      *
000037  ;                       *
000038  ;     NBUF1 AND NBUF2   *
000039  ;       HOLD 6-BIT NIBLS*
000040  ;       (00ABCDEF)      *
000041  ;                       *
000042  ;     USES TEMP 'IDX'.  *
000043  ;                       *
000044  ;   ---- CAUTION -----   *
000045  ;                       *
000046  ;        OBSERVE        *
000047  ;    'NO PAGE CROSS'    *
000048  ;      WARNINGS ON      *
000049  ;    SOME BRANCHES!!    *
000050  ;                       *
000051  ;   ---- ASSUMES ----   *
000052  ;                       *
000053  ;   1 USEC CYCLE TIME   *
000054  ;                       *
000055  ;************************
000056  READ16          LDY          #020                    ;'MUST FIND' COUNT.
000057  RSYNC           DEY                                  ;IF CAN'T FIND MARKS
000058                  BEQ          RDERR                   ;THEN EXIT WITH CARRY SET.
000059  RD1             LDA          Q6L,X                   ;READ NIBL.
000060                  BPL          RD1                     ;**; NO PAGE CROSS! ***
000061  RSYNC1          EOR          #0D5                    ;DATA MARK 1?
000062                  BNE          RSYNC                   ;LOOP IF NOT.
000063                  NOP                                  ;DELAY BETWEEN NIBLS.
000064  RD2             LDA          Q6L,X
000065                  BPL          RD2                     ;**; NO PAGE CROSS! ***
000066                  CMP          #0AA                    ;DATA MARK 2?
000067                  BNE          RSYNC1                  ;(IF NOT, IS IT DM1?)
000068                  LDY          #056                    ;INIT NBUF2 INDEX.
000069  ;              (ADDED NIBL DELAY)
000070  RD3             LDA          Q6L,X
000071                  BPL          RD3                     ;**; NO PAGE CROSS! ***
000072                  CMP          #0AD                    ;DATA MARK 3?
000073                  BNE          RSYNC1                  ;(IF NOT, IS IT DM1?)
000074  ;         (CARRY SET IF DM3!)
000075                  LDA          #0                      ;INIT CHECKSUM.
000076  RDATA1          DEY
```

```
000077                  STY         IDX
000078  RD4             LDY         Q6L,X
000079                  BPL         RD4                     ;**; NO PAGE CROSS! ***
000080  RDNIBL1         EOR         DNIBL,Y                 ;XOR 6-BIT NIBL.
000081                  LDY         IDX
000082
000083  ; NOTE: The storing of the Nybble into the NBUF has been deleted so that
000084  ;       the formatter doesn't have to prenibblize the data pattern again.
000085
000086  ;               STA     NBUF2,Y                 ;STORE IN NBUF2 PAGE.
000087
000088                  BNE         RDATA1                  ;TAKEN IF Y-REG NONZERO.
000089  RDATA2          STY         IDX
000090  RD5             LDY         Q6L,X
000091                  BPL         RD5                     ;**; NO PAGE CROSS! ***
000092  RDNIBL2         EOR         DNIBL,Y                 ;XOR 6-BIT NIBL.
000093                  LDY         IDX
000094
000095  ; NOTE: The storing of the Nybble into the NBUF has been deleted so that
000096  ;       the formatter doesn't have to prenibblize the data pattern again.
000097
000098  ;               STA     NBUF1,Y                 ;STORE IN NBUF1 PAGE.
000099
000100                  INY
000101                  BNE         RDATA2
000102  RD6             LDY         Q6L,X                   ;READ 7-BIT CSUM NIBL.
000103                  BPL         RD6                     ;**; NO PAGE CROSS! ***
000104  RDNIBL3         CMP         DNIBL,Y                 ;IF LAST NBUF1 NIBL NOT
000105                  BNE         RDERR                   ;EQUAL CHKSUM NIBL THEN ERR.
000106  RD7             LDA         Q6L,X
000107                  BPL         RD7                     ;**; NO PAGE CROSS! ***
000108                  CMP         #0DE                    ;FIRST BIT SLIP MARK?
000109                  BNE         RDERR                   ;(ERR IF NOT)
000110                  NOP                                 ;DELAY BETWEEN NIBLS.
000111  RD8             LDA         Q6L,X
000112                  BPL         RD8                     ;**; NO PAGE CROSS! ***
000113                  CMP         #0AA                    ;SECOND BIT SLIP MARK?
000114                  BEQ         RDEXIT                  ;(DONE IF IT IS)
000115  RDERR           SEC                                 ;INDICATE 'ERROR EXIT'.
000116                  RTS                                 ;RETURN FROM READ16 OR RDADR16.
000117  RDEXIT          CLC
000118                  RTS

; ##############################################################################################################
; #   END OF FILE:  FMTREAD.TEXT
; #   LINES      :  118
; #   CHARACTERS :  5863
; #   Formatter  :  Assembly Language Reformatter 1.0.2 (06 January 1998)
; #   Author     :  David T. Craig -- 71533.606@compuserve.com -- Santa Fe, New Mexico USA
; ##############################################################################################################
```

```
; ##########################################################################################################
; #   PROJECT  :  Apple /// SOS DISK /// Formatter Device Driver 1.30 (6502 Assembly Source Code)
; #   FILE NAME:  FMTWRADDR.TEXT
; ##########################################################################################################

000001                  .TITLE       "16-SECTOR WRITE ADDRESS"
000002                  .PAGE
000003  ;*******************************
000004  ;                              *
000005  ;  WRITE ADR FIELD SUBROUTINE  *
000006  ;      (16-SECTOR FORMAT)      *
000007  ;  WRITES SPECIFIED NUMBER OF  *
000008  ;  40-USEC (10-BIT) SELF-SYNC  *
000009  ;  NIBLS, ADR FIELDS 16-SECTOR *
000010  ;  START MARKS ($D5,$AA,$96),  *
000011  ;  BODY (VOLUME, TRACK, SECTOR,*
000012  ;  CHECKSUM), END FIELD MARKS, *
000013  ;  AND THE WRITE TURN-OFF NIBL.*
000014  ;                              *
000015  ;*******************************
000016  ;                              *
000017  ;   ------- ON ENTRY -------   *
000018  ;                              *
000019  ;  THE LOCATIONS VOLUME, TRK,  *
000020  ;  AND NSECT MUST CONTAIN THE  *
000021  ;  DESIRED VOLUME, TRACK, AND  *
000022  ;  SECTOR VALUES DESIRED.      *
000023  ;                              *
000024  ;  THE PROPER DRIVE MUST BE    *
000025  ;  ENABLED AND UP TO SPEED IN  *
000026  ;  READ MODE (Q7L, Q6L).       *
000027  ;                              *
000028  ;  X-REG CONTAINS SLOTNUM      *
000029  ;                    TIMES 16. *
000030  ;                              *
000031  ;  Y-REG CONTAINS NUMBER OF    *
000032  ;    SELF-SYNC NIBLS DESIRED   *
000033  ;    MINUS 1.                  *
000034  ;      (0 FOR 256 NIBLS)       *
000035  ;                              *
000036  ;*******************************
000037  ;                              *
000038  ;   ------- REQUIRES -------   *
000039  ;                              *
000040  ;         1 USEC CYCLE         *
000041  ;                              *
000042  ;*******************************
000043  ;                              *
000044  ;   ------- CAUTION --------   *
000045  ;                              *
000046  ;  MOST OF THIS CODE IS TIME   *
000047  ;  CRITICAL.  OBSERVE ALL      *
000048  ;  'NO PAGE CROSS!' WARNINGS   *
000049  ;  ON BRANCHES.                *
000050  ;                              *
000051  ;*******************************
000052  WADR16          SEC                                 ;ANTICIPATE WR PROT ERR.
000053                  LDA          Q6H,X                  ;INTO 'WR PROT SENSE' MODE.
000054                  LDA          Q7L,X                  ;SENSE IT (NEG=PROTECTED)
000055                  BMI          WADRTS                 ;ERR EXIT IF PROTECTED.
000056                  LDA          #0FF                   ;SELF-SYNC NIBL.
000057                  STA          Q7H,X                  ;WRITE FIRST NIBL.
000058                  CMP          Q6L,X                  ;(4) BACK TO WRITE MODE.
000059                  PHA                                 ;(3) FOR DELAY.
000060                  PLA                                 ;(4)
000061  WSYNC1          JSR          WADRTS1                ;(12) FOR 40-USEC NIBLS.
000062                  JSR          WADRTS1                ;(12)
000063                  STA          Q6H,X                  ;(5) WRITE NIBL.
000064                  CMP          Q6L,X                  ;(4) (BACK TO WRITE MODE)
000065                  NOP                                 ;(2) FOR DELAY.
000066                  DEY                                 ;(2) NEXT OF 'N' NIBLS.
000067                  BNE          WSYNC1                 ;(3) **; NO PAGE CROSS! ***
000068                  LDA          #0D5                   ;(2) ADR MARK 1.
000069                  JSR          WNIBLB2                ;(15,9,6) WRITE IT.
000070                  LDA          #0AA                   ;(2) ADR MARK 2.
000071                  JSR          WNIBLB2                ;(15,9,6) WRITE IT.
000072                  LDA          #096                   ;(2) 16-SECTOR ADR MARK 3.
000073                  JSR          WNIBLB2                ;(15,9,6) WRITE IT.
000074                  LDA          NVOL                   ;(3)
000075
              JSR          WBYTE                  ;(14,9,6) WRITE NVOL (ODD, THEN EVEN, BITS.)
```

```
000157                  LDA         Q6H,X                   ;SENSE WRITEPROT MODE
000158                  LDA         Q7L,X
000159                  TAY                                 ;SAVE RESULT
000160                  LDA         Q7L,X                   ;BACK INTO
000161                  LDA         Q6L,X                   ; READ MODE
000162                  TYA
000163                  BPL         $010                    ;IT'S OK.
000164                  LDA         #2
000165                  JMP         FORMDONE                ;FLAG WRITE PROTECT.
000166
000167  $010            JSR         WT16                    ;WRITE AND VERIFY TRACK.    /RRA82237/
000168                  BCC         $020                    ;=>NO ERROR
000169
000170  ; BEFORE SAYING IT'S BAD MEDIA, SEE IF THE DRIVE'S WAY TOO FAST:      /RRA82237/
000171
000172                  LDA         #1                      ;ASSUME BAD MEDIA     /RRA82237/
000173                  LDY         NSYNC                   ;GET IT               /RRA82237/
000174                  CPY         DCB_MINSYNC             ;GAPS TOO SMALL?      /RRA82237/
000175                  BCS         $015                    ;GAPS OK: BAD MEDIA   /RRA82237/
000176                  LDA         #4                      ;"DRIVE TOO FAST"     /RRA82237/
000177  $015            JMP         FORMDONE
000178
000179  ; THE MEDIA FORMATTED OK...BUT THE DRIVE MIGHT STILL BE TOO SLOW OR TOO FAST.
000180
000181  $020            LDY         NSYNC                   ;GET IT               /RRA82237/
000182                  CPY         DCB_MINSYNC             ;GAPS TOO SMALL?      /RRA82237/
000183                  BCS         $025                    ;GAPS LARGE ENOUGH    /RRA82237/
000184                  LDA         #4                      ;"DRIVE TOO FAST"     /RRA82237/
000185                  JMP         FORMDONE                ;                     /RRA82237/
000186  $025            CPY         DCB_MAXSYNC             ;GAPS TOO LARGE?      /RRA82237/
000187                  BCC         $030                    ;->OK                 /RRA82237/
000188                  LDA         #3                      ;"DRIVE TOO SLOW"     /RRA82237/
000189                  JMP         FORMDONE                ;                     /RRA82237/
000190  $030            LDA         DCB_SECTRETRY           ;SECTOR RETRIES       /RRA82258/
000191                  STA         RETRYCNT                ; TO FIND SECTOR 0.
000192  FINDS0          .EQU        *
000193                  DEC         RETRYCNT                ;DONE RETRIES?
000194                  BNE         $010                    ;=>NO, KEEP RETRYING.
000195                  LDA         #1                      ;'UNABLE TO FORMAT' ERROR
000196                  JMP         FORMDONE
000197
000198  $010            LDX         #60
000199                  JSR         RDADR16                 ;READ ADR FIELD.
000200                  BCS         FINDS0                  ;RETRY IF ERR.
000201                  LDA         SECT                    ;CHECK SECTOR THAT WAS READ.
000202                  BNE         FINDS0                  ;CONTINUE SEARCHING IF NOT SECT 0.
000203                  LDX         #60
000204                  JSR         READ16                  ;NOW READ DATA FIELD.
000205                  BCS         FINDS0                  ;CONTINUE SEARCH IF ERR.
000206  ;       (NOW POSITIONED PROPERLY FOR NEXT TRACK)
000207                  INC         TRK                     ;INCREMENT TRACK NUMBER.
000208                  LDA         TRK
000209                  CMP         #35.                    ;CONTINUE IF LESS THAN 35.
000210                  BCC         FORMTRK
000211
000212                  LDA         #0                      ;GOOD COMPLETION
000213
000214  FORMDONE        PHA                                 ;SAVE RETURN CODE
000215                  LDX         #60
000216                  LDA         MOTOROFF,X              ;TURN MOTOR OFF.
000217                  LDA         #0
000218                  JSR         MYSEEK                  ;RESTORE HEAD TO TRACK 00
000219
000220  ; RESTORE THE ROM'S ZEROPAGE AREA:
000221
000222                  LDX         #31.                    ;INDEX
000223  $035            LDA         XTNDSAVE,X              ;RESTORE EXTEND
000224                  STA         1480,X
000225                  LDA         ZEROSAVE,X
000226                  STA         80,X
000227                  DEX
000228                  BPL         $035
000229
000230  ; Restore the caller's Environment:
000231
000232                  LDA         P_SAVE                  ;RESTORE
000233                  PHA                                 ; INHIBITS
000234                  PLP
000235                  LDA         E_SAVE                  ;RESTORE
000236                  STA         E_REG                   ; ENVIRONMENT
000237
```

```
000238                  PLA                                     ;RESTORE RETURN CODE
000239                  RTS                                     ; AND RETURN.

; ##############################################################################################################
; #   END OF FILE:  FMTFORMAT.TEXT
; #   LINES      :  239
; #   CHARACTERS :  13436
; #   Formatter  :  Assembly Language Reformatter 1.0.2 (06 January 1998)
; #   Author     :  David T. Craig -- 71533.606@compuserve.com -- Santa Fe, New Mexico USA
; ##############################################################################################################
```

```
000158
000159  ; Delay to let most of sector 15's data pass by, then go back and     /RRA82258/
000160  ;  rewrite the track (including the 128-nibbles of seam). This assures/RRA82258/
000161  ;  that we will rewrite the track entirely, and that sector 0 will be /RRA82258/
000162  ;  aligned at roughly the same place as it was on the prior try.      /RRA82258/
000163
000164  PASS15          .EQU        *                       ;                       /RRA82258/
000165                  LDX         #342.-128.              ;LET MOST GO BY         /RRA82258/
000166  VFYDELAY        .EQU        *                       ;                       /RRA82258/
000167                  JSR         WEXIT2                  ;(12)                   /RRA82258/
000168                  JSR         WEXIT2                  ;(12)                   /RRA82258/
000169                  BIT         0                       ;( 3)                   /RRA82258/
000170                  DEX                                 ;( 2)                   /RRA82258/
000171                  BNE         VFYDELAY                ;( 3)                   /RRA82258/
000172                  JMP         WTRETRY                 ;WRITE TRACK FROM HERE IF NO ERR.

; ###############################################################################################################
; #   END OF FILE:  FMTWTRACK.TEXT
; #   LINES      :  172
; #   CHARACTERS :  10797
; #   Formatter  :  Assembly Language Reformatter 1.0.2 (06 January 1998)
; #   Author     :  David T. Craig -- 71533.606@compuserve.com -- Santa Fe, New Mexico USA
; ###############################################################################################################

###
```