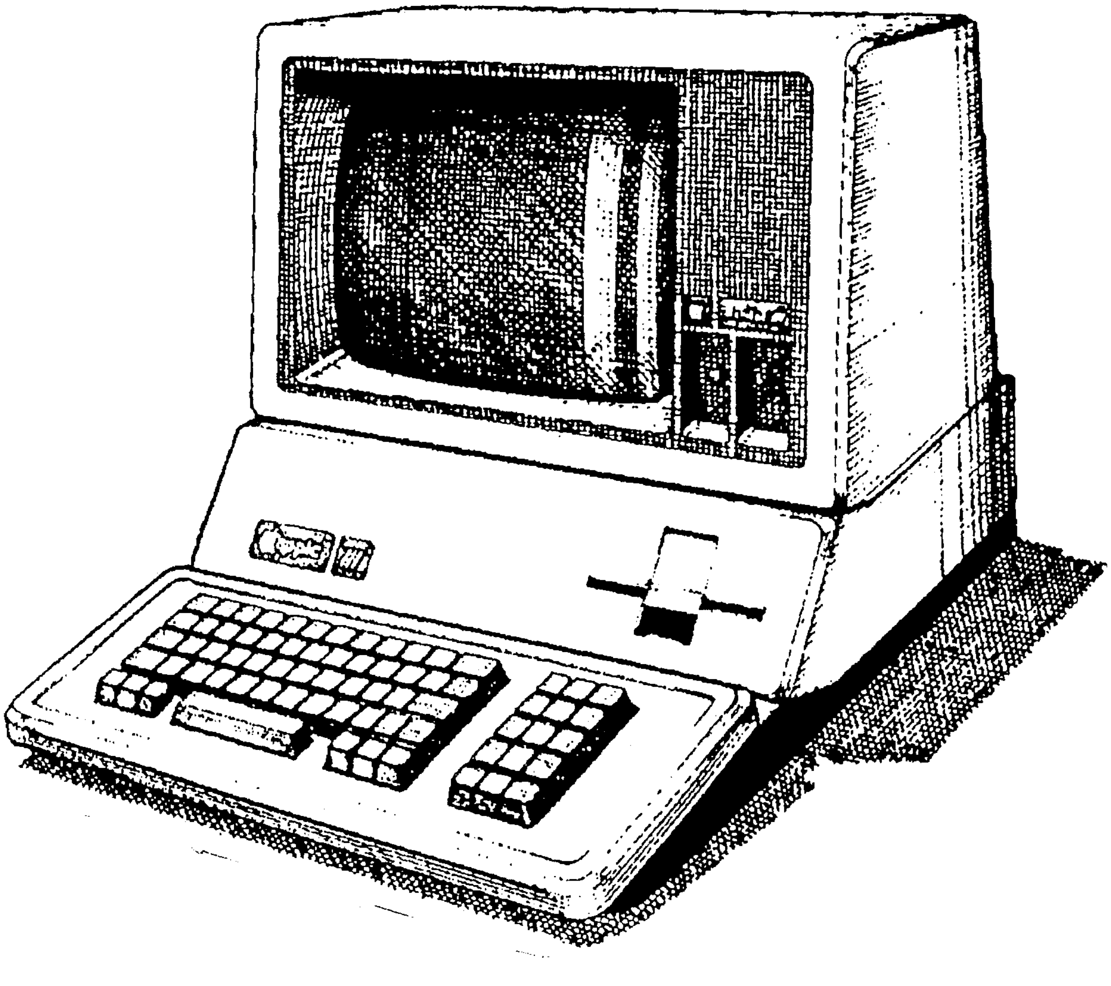
Apple /// Computer Information



DOCUMENT NAME

APPLE /// APPLE][EMULATION
MONITOR SOURCE CODE LISTING

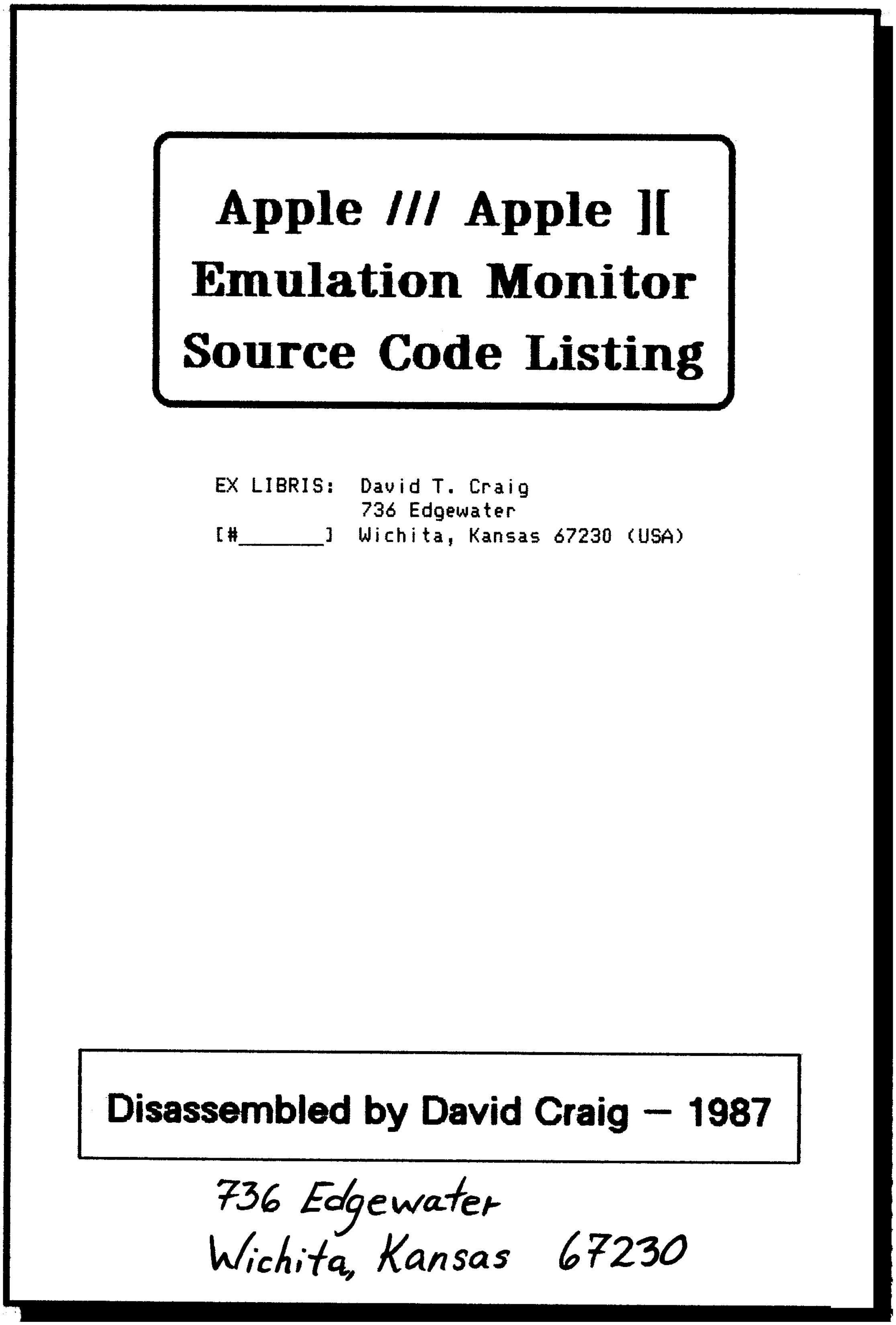
#

90

Ex Libris David T. Craig

"_03.PICT" 241 KB 2001-08-13 dpi: 300h x 300v pix: 2166h x 2941v

45 pages

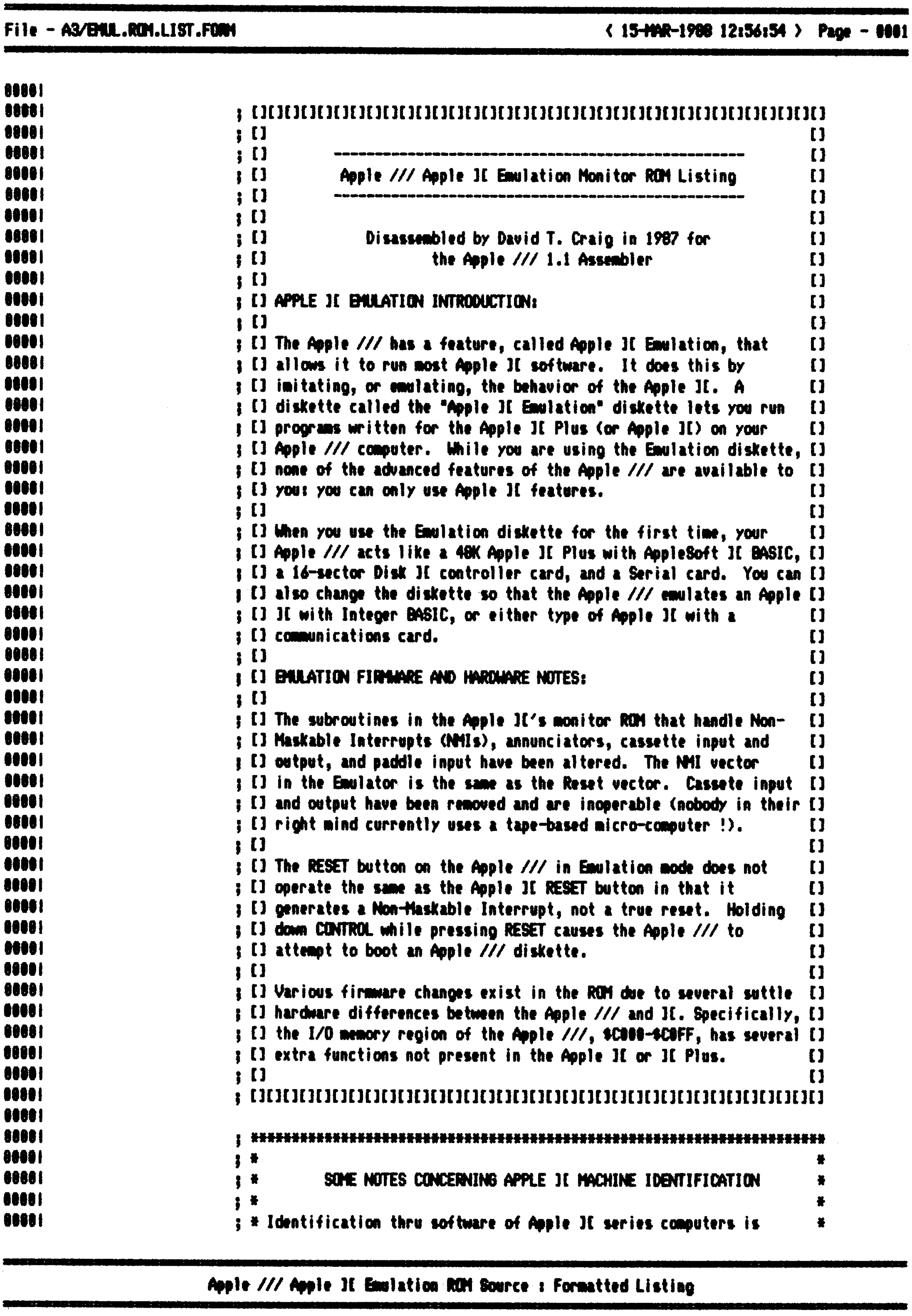


Apple /// Apple][Emulation Monitor Source Code Listing

EX LIBRIS: David T. Craig
736 Edgewater
[#________] Wichita, Kansas 67230 (USA)

Disassembled by David Craig — 1987

736 Edgewater
Wichita, Kansas 67230

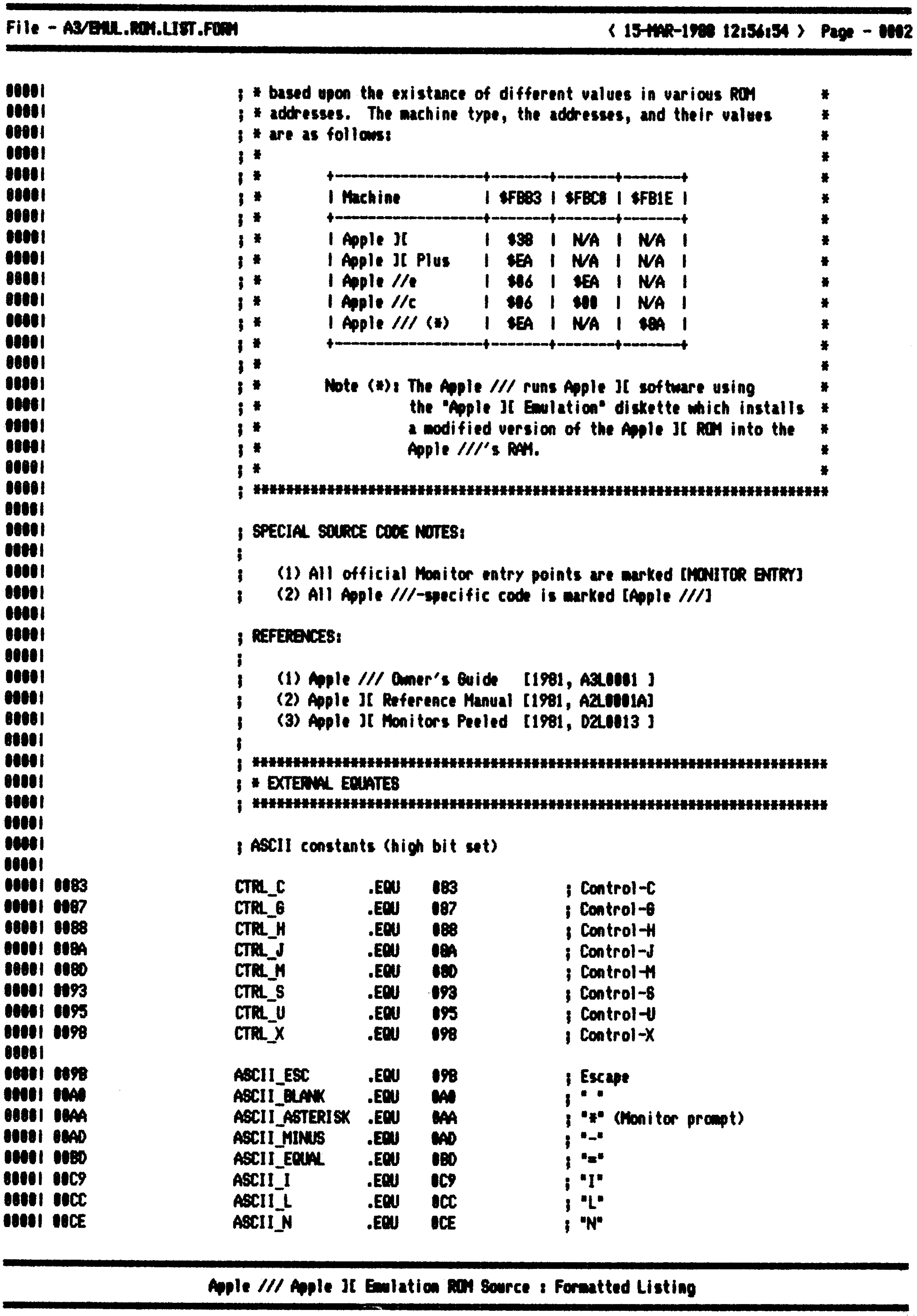


File - A3/EMUL.ROM.LIST.FORM   < 15-MAR-1988 12:56:54 > Page - 0001

```
0000|
0000|                    ; [][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][]
0000|                    ; []                                                                  []
0000|                    ; []       ------------------------------------------------           []
0000|                    ; []        Apple /// Apple ][ Emulation Monitor ROM Listing          []
0000|                    ; []       ------------------------------------------------           []
0000|                    ; []                                                                  []
0000|                    ; []           Disassembled by David T. Craig in 1987 for             []
0000|                    ; []                  the Apple /// 1.1 Assembler                     []
0000|                    ; []                                                                  []
0000|                    ; [] APPLE ][ EMULATION INTRODUCTION:                                 []
0000|                    ; []                                                                  []
0000|                    ; [] The Apple /// has a feature, called Apple ][ Emulation, that     []
0000|                    ; [] allows it to run most Apple ][ software.  It does this by        []
0000|                    ; [] imitating, or emulating, the behavior of the Apple ][.  A        []
0000|                    ; [] diskette called the "Apple ][ Emulation" diskette lets you run   []
0000|                    ; [] programs written for the Apple ][ Plus (or Apple ][) on your     []
0000|                    ; [] Apple /// computer.  While you are using the Emulation diskette, []
0000|                    ; [] none of the advanced features of the Apple /// are available to  []
0000|                    ; [] you: you can only use Apple ][ features.                         []
0000|                    ; []                                                                  []
0000|                    ; [] When you use the Emulation diskette for the first time, your     []
0000|                    ; [] Apple /// acts like a 48K Apple ][ Plus with AppleSoft ][ BASIC, []
0000|                    ; [] a 16-sector Disk ][ controller card, and a Serial card.  You can []
0000|                    ; [] also change the diskette so that the Apple /// emulates an Apple []
0000|                    ; [] ][ with Integer BASIC, or either type of Apple ][ with a         []
0000|                    ; [] communications card.                                             []
0000|                    ; []                                                                  []
0000|                    ; [] EMULATION FIRMWARE AND HARDWARE NOTES:                           []
0000|                    ; []                                                                  []
0000|                    ; [] The subroutines in the Apple ]['s monitor ROM that handle Non-   []
0000|                    ; [] Maskable Interrupts (NMIs), annunciators, cassette input and     []
0000|                    ; [] output, and paddle input have been altered.  The NMI vector      []
0000|                    ; [] in the Emulator is the same as the Reset vector.  Cassete input  []
0000|                    ; [] and output have been removed and are inoperable (nobody in their []
0000|                    ; [] right mind currently uses a tape-based micro-computer !).        []
0000|                    ; []                                                                  []
0000|                    ; [] The RESET button on the Apple /// in Emulation mode does not     []
0000|                    ; [] operate the same as the Apple ][ RESET button in that it         []
0000|                    ; [] generates a Non-Maskable Interrupt, not a true reset.  Holding   []
0000|                    ; [] down CONTROL while pressing RESET causes the Apple /// to        []
0000|                    ; [] attempt to boot an Apple /// diskette.                           []
0000|                    ; []                                                                  []
0000|                    ; [] Various firmware changes exist in the ROM due to several suttle  []
0000|                    ; [] hardware differences between the Apple /// and ][. Specifically, []
0000|                    ; [] the I/O memory region of the Apple ///, $C000-$C0FF, has several []
0000|                    ; [] extra functions not present in the Apple ][ or ][ Plus.          []
0000|                    ; []                                                                  []
0000|                    ; [][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][]
0000|
0000|                    ; **********************************************************************
0000|                    ; *                                                                    *
0000|                    ; *         SOME NOTES CONCERNING APPLE ][ MACHINE IDENTIFICATION      *
0000|                    ; *                                                                    *
0000|                    ; * Identification thru software of Apple ][ series computers is       *
```

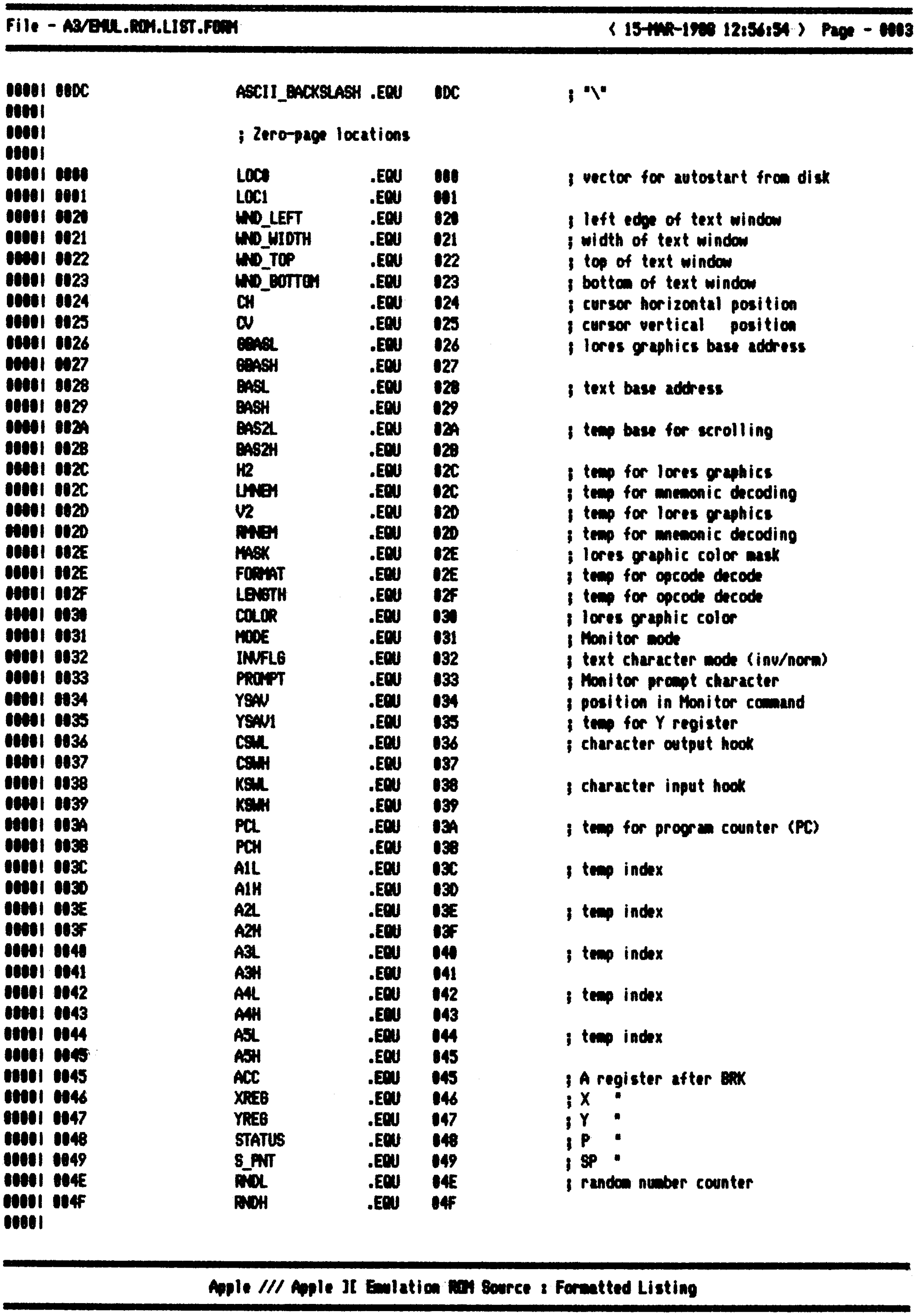
Apple /// Apple][Emulation ROM Source : Formatted Listing

“_05.PICT” 252 KB 2001-08-13 dpi: 300h x 300v pix: 2237h x 3166v



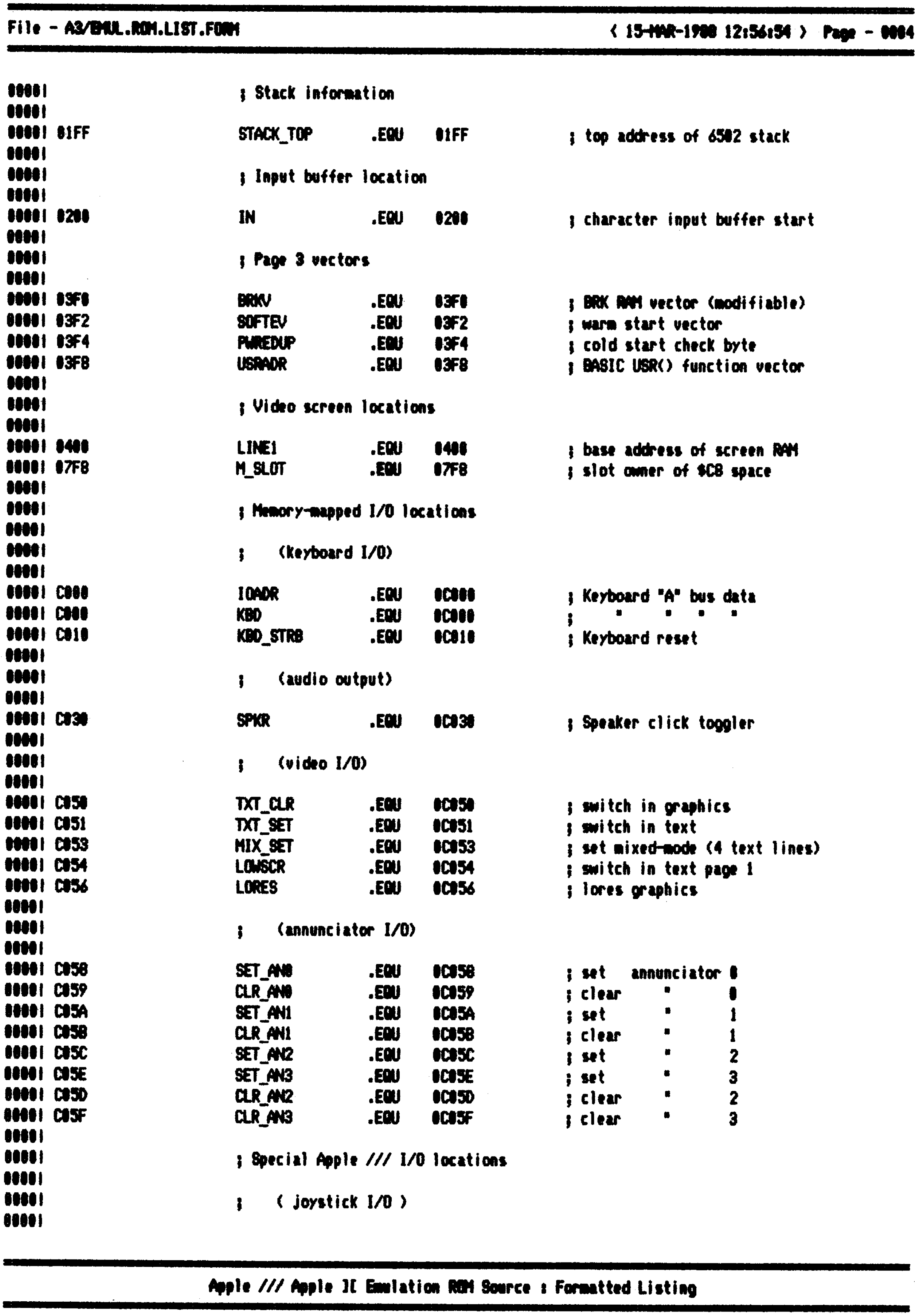
```
0000|                        ; * based upon the existance of different values in various ROM      *
0000|                        ; * addresses.  The machine type, the addresses, and their values    *
0000|                        ; * are as follows:                                                  *
0000|                        ; *                                                                  *
0000|                        ; *       +-----------------+-------+-------+-------+                *
0000|                        ; *       | Machine         | $FBB3 | $FBC0 | $FB1E |                *
0000|                        ; *       +-----------------+-------+-------+-------+                *
0000|                        ; *       | Apple ][        |  $38  |  N/A  |  N/A  |                *
0000|                        ; *       | Apple ][ Plus   |  $EA  |  N/A  |  N/A  |                *
0000|                        ; *       | Apple //e       |  $06  |  $EA  |  N/A  |                *
0000|                        ; *       | Apple //c       |  $06  |  $00  |  N/A  |                *
0000|                        ; *       | Apple /// (*)   |  $EA  |  N/A  |  $8A  |                *
0000|                        ; *       +-----------------+-------+-------+-------+                *
0000|                        ; *                                                                  *
0000|                        ; *       Note (*): The Apple /// runs Apple ][ software using       *
0000|                        ; *                 the "Apple ][ Emulation" diskette which installs *
0000|                        ; *                 a modified version of the Apple ][ ROM into the  *
0000|                        ; *                 Apple ///'s RAM.                                 *
0000|                        ; *                                                                  *
0000|                        ; ********************************************************************
0000|
0000|                        ; SPECIAL SOURCE CODE NOTES:
0000|                        ;
0000|                        ;   (1) All official Monitor entry points are marked [MONITOR ENTRY]
0000|                        ;   (2) All Apple ///-specific code is marked [Apple ///]
0000|
0000|                        ; REFERENCES:
0000|                        ;
0000|                        ;   (1) Apple /// Owner's Guide   [1981, A3L0001 ]
0000|                        ;   (2) Apple ][ Reference Manual [1981, A2L0001A]
0000|                        ;   (3) Apple ][ Monitors Peeled  [1981, D2L0013 ]
0000|                        ;
0000|                        ; ********************************************************************
0000|                        ; * EXTERNAL EQUATES
0000|                        ; ********************************************************************
0000|
0000|                        ; ASCII constants (high bit set)
0000|
0000| 0083                   CTRL_C          .EQU    083             ; Control-C
0000| 0087                   CTRL_G          .EQU    087             ; Control-G
0000| 0088                   CTRL_H          .EQU    088             ; Control-H
0000| 008A                   CTRL_J          .EQU    08A             ; Control-J
0000| 008D                   CTRL_M          .EQU    08D             ; Control-M
0000| 0093                   CTRL_S          .EQU    093             ; Control-S
0000| 0095                   CTRL_U          .EQU    095             ; Control-U
0000| 0098                   CTRL_X          .EQU    098             ; Control-X
0000|
0000| 009B                   ASCII_ESC       .EQU    09B             ; Escape
0000| 00A0                   ASCII_BLANK     .EQU    0A0             ; " "
0000| 00AA                   ASCII_ASTERISK  .EQU    0AA             ; "*" (Monitor prompt)
0000| 00AD                   ASCII_MINUS     .EQU    0AD             ; "-"
0000| 00BD                   ASCII_EQUAL     .EQU    0BD             ; "="
0000| 00C9                   ASCII_I         .EQU    0C9             ; "I"
0000| 00CC                   ASCII_L         .EQU    0CC             ; "L"
0000| 00CE                   ASCII_N         .EQU    0CE             ; "N"
```

"_06.PICT" 190 KB 2001-08-13 dpi: 300h x 300v pix: 2219h x 3166v



```
0000| 00DC               ASCII_BACKSLASH .EQU    0DC            ; "\"
0000|
0000|                    ; Zero-page locations
0000|
0000| 0000               LOC0            .EQU    000            ; vector for autostart from disk
0000| 0001               LOC1            .EQU    001
0000| 0020               WND_LEFT        .EQU    020            ; left edge of text window
0000| 0021               WND_WIDTH       .EQU    021            ; width of text window
0000| 0022               WND_TOP         .EQU    022            ; top of text window
0000| 0023               WND_BOTTOM      .EQU    023            ; bottom of text window
0000| 0024               CH              .EQU    024            ; cursor horizontal position
0000| 0025               CV              .EQU    025            ; cursor vertical   position
0000| 0026               GBASL           .EQU    026            ; lores graphics base address
0000| 0027               GBASH           .EQU    027
0000| 0028               BASL            .EQU    028            ; text base address
0000| 0029               BASH            .EQU    029
0000| 002A               BAS2L           .EQU    02A            ; temp base for scrolling
0000| 002B               BAS2H           .EQU    02B
0000| 002C               H2              .EQU    02C            ; temp for lores graphics
0000| 002C               LMNEM           .EQU    02C            ; temp for mnemonic decoding
0000| 002D               V2              .EQU    02D            ; temp for lores graphics
0000| 002D               RMNEM           .EQU    02D            ; temp for mnemonic decoding
0000| 002E               MASK            .EQU    02E            ; lores graphic color mask
0000| 002E               FORMAT          .EQU    02E            ; temp for opcode decode
0000| 002F               LENGTH          .EQU    02F            ; temp for opcode decode
0000| 0030               COLOR           .EQU    030            ; lores graphic color
0000| 0031               MODE            .EQU    031            ; Monitor mode
0000| 0032               INVFLG          .EQU    032            ; text character mode (inv/norm)
0000| 0033               PROMPT          .EQU    033            ; Monitor prompt character
0000| 0034               YSAV            .EQU    034            ; position in Monitor command
0000| 0035               YSAV1           .EQU    035            ; temp for Y register
0000| 0036               CSWL            .EQU    036            ; character output hook
0000| 0037               CSWH            .EQU    037
0000| 0038               KSWL            .EQU    038            ; character input hook
0000| 0039               KSWH            .EQU    039
0000| 003A               PCL             .EQU    03A            ; temp for program counter (PC)
0000| 003B               PCH             .EQU    03B
0000| 003C               A1L             .EQU    03C            ; temp index
0000| 003D               A1H             .EQU    03D
0000| 003E               A2L             .EQU    03E            ; temp index
0000| 003F               A2H             .EQU    03F
0000| 0040               A3L             .EQU    040            ; temp index
0000| 0041               A3H             .EQU    041
0000| 0042               A4L             .EQU    042            ; temp index
0000| 0043               A4H             .EQU    043
0000| 0044               A5L             .EQU    044            ; temp index
0000| 0045               A5H             .EQU    045
0000| 0045               ACC             .EQU    045            ; A register after BRK
0000| 0046               XREG            .EQU    046            ; X   "
0000| 0047               YREG            .EQU    047            ; Y   "
0000| 0048               STATUS          .EQU    048            ; P   "
0000| 0049               S_PNT           .EQU    049            ; SP  "
0000| 004E               RNDL            .EQU    04E            ; random number counter
0000| 004F               RNDH            .EQU    04F
0000|
```

“_07.PICT” 199 KB 2001-08-13 dpi: 300h x 300v pix: 2219h x 3178v

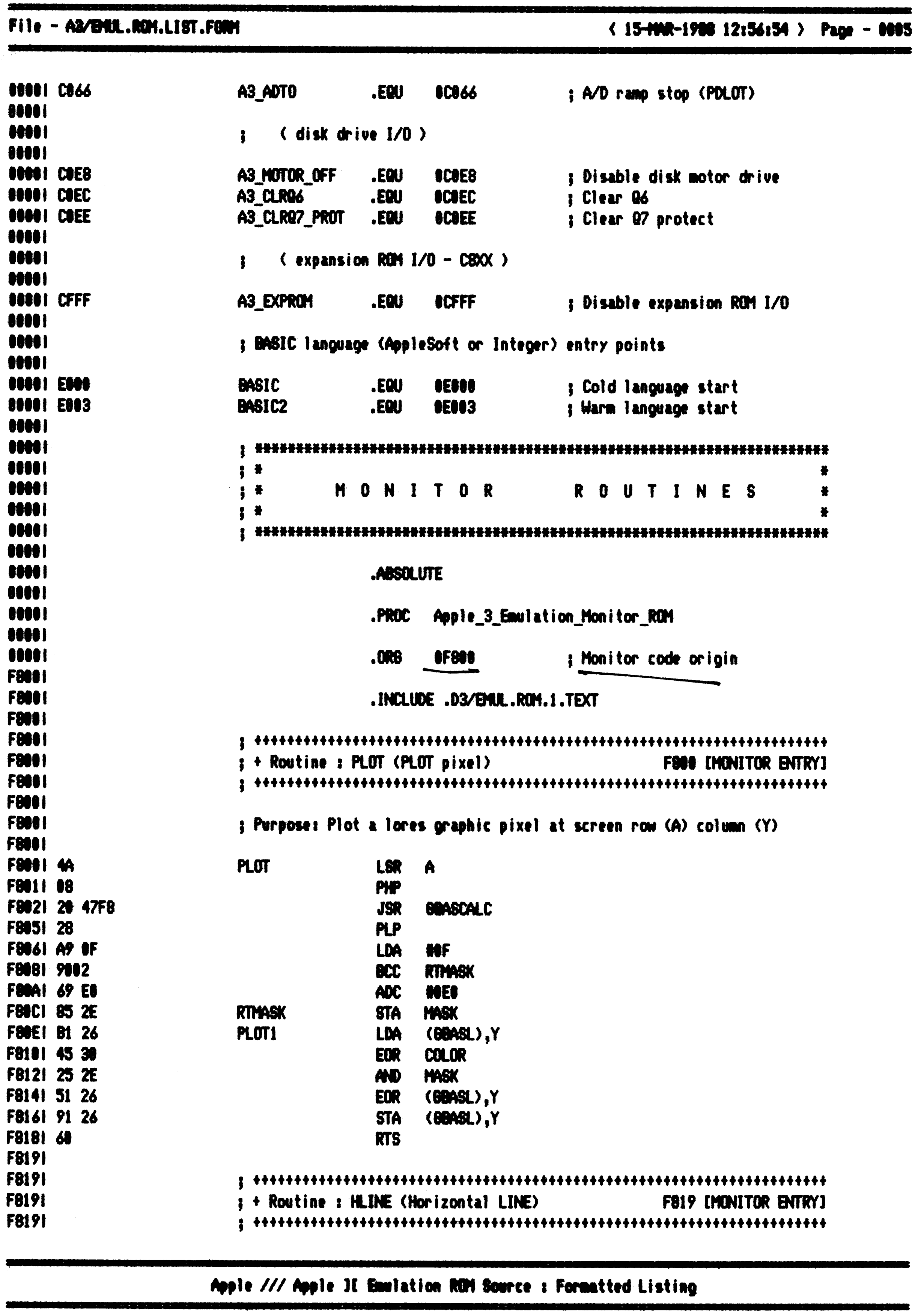


File - A3/EMUL.ROM.LIST.FORM (15-MAR-1980 12:56:54) Page - 0004

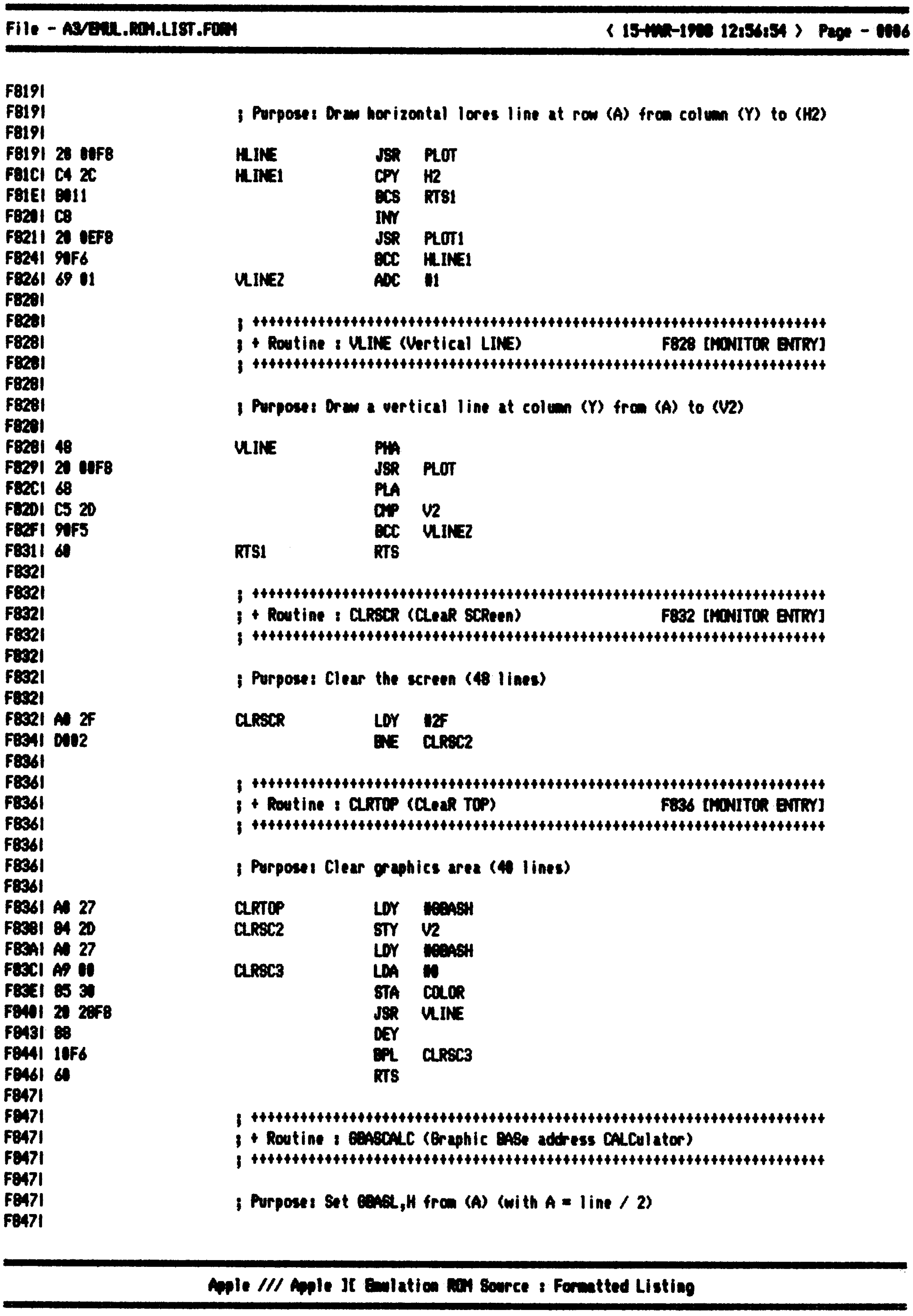
```
0000|                     ; Stack information
0000|
0000| 01FF                STACK_TOP      .EQU    01FF          ; top address of 6502 stack
0000|
0000|                     ; Input buffer location
0000|
0000| 0200                IN             .EQU    0200          ; character input buffer start
0000|
0000|                     ; Page 3 vectors
0000|
0000| 03F0                BRKV           .EQU    03F0          ; BRK RAM vector (modifiable)
0000| 03F2                SOFTEV         .EQU    03F2          ; warm start vector
0000| 03F4                PWREDUP        .EQU    03F4          ; cold start check byte
0000| 03F8                USRADR         .EQU    03F8          ; BASIC USR() function vector
0000|
0000|                     ; Video screen locations
0000|
0000| 0400                LINE1          .EQU    0400          ; base address of screen RAM
0000| 07F8                M_SLOT         .EQU    07F8          ; slot owner of $C8 space
0000|
0000|                     ; Memory-mapped I/O locations
0000|
0000|                     ;    (Keyboard I/O)
0000|
0000| C000                IOADR          .EQU    0C000         ; Keyboard "A" bus data
0000| C000                KBD            .EQU    0C000         ;    "    "   "   "
0000| C010                KBD_STRB       .EQU    0C010         ; Keyboard reset
0000|
0000|                     ;    (audio output)
0000|
0000| C030                SPKR           .EQU    0C030         ; Speaker click toggler
0000|
0000|                     ;    (video I/O)
0000|
0000| C050                TXT_CLR        .EQU    0C050         ; switch in graphics
0000| C051                TXT_SET        .EQU    0C051         ; switch in text
0000| C053                MIX_SET        .EQU    0C053         ; set mixed-mode (4 text lines)
0000| C054                LOWSCR         .EQU    0C054         ; switch in text page 1
0000| C056                LORES          .EQU    0C056         ; lores graphics
0000|
0000|                     ;    (annunciator I/O)
0000|
0000| C058                SET_AN0        .EQU    0C058         ; set   annunciator 0
0000| C059                CLR_AN0        .EQU    0C059         ; clear      "      0
0000| C05A                SET_AN1        .EQU    0C05A         ; set        "      1
0000| C05B                CLR_AN1        .EQU    0C05B         ; clear      "      1
0000| C05C                SET_AN2        .EQU    0C05C         ; set        "      2
0000| C05E                SET_AN3        .EQU    0C05E         ; set        "      3
0000| C05D                CLR_AN2        .EQU    0C05D         ; clear      "      2
0000| C05F                CLR_AN3        .EQU    0C05F         ; clear      "      3
0000|
0000|                     ; Special Apple /// I/O locations
0000|
0000|                     ;    ( joystick I/O )
0000|
```

Apple /// Apple][Emulation ROM Source : Formatted Listing

"_08.PICT" 159 KB 2001-08-13 dpi: 300h x 300v pix: 2237h x 3178v



```
File - A3/EMUL.ROM.LIST.FORM                                        < 15-MAR-1988 12:56:54 >  Page - 0005

0000| C066              A3_ADTO         .EQU    $C066           ; A/D ramp stop (PDLOT)
0000|
0000|                   ;    ( disk drive I/O )
0000|
0000| C0E8              A3_MOTOR_OFF    .EQU    $C0E8           ; Disable disk motor drive
0000| C0EC              A3_CLRQ6        .EQU    $C0EC           ; Clear Q6
0000| C0EE              A3_CLRQ7_PROT   .EQU    $C0EE           ; Clear Q7 protect
0000|
0000|                   ;    ( expansion ROM I/O - C8XX )
0000|
0000| CFFF              A3_EXPROM       .EQU    $CFFF           ; Disable expansion ROM I/O
0000|
0000|                   ; BASIC language (AppleSoft or Integer) entry points
0000|
0000| E000              BASIC           .EQU    $E000           ; Cold language start
0000| E003              BASIC2          .EQU    $E003           ; Warm language start
0000|
0000|                   ; ******************************************************************
0000|                   ; *                                                                *
0000|                   ; *       M O N I T O R        R O U T I N E S                     *
0000|                   ; *                                                                *
0000|                   ; ******************************************************************
0000|
0000|                                   .ABSOLUTE
0000|
0000|                                   .PROC   Apple_3_Emulation_Monitor_ROM
0000|
0000|                                   .ORG    $F800           ; Monitor code origin
F800|
F800|                                   .INCLUDE .D3/EMUL.ROM.1.TEXT
F800|
F800|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F800|                   ; + Routine : PLOT (PLOT pixel)                   F800 [MONITOR ENTRY]
F800|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F800|
F800|                   ; Purpose: Plot a lores graphic pixel at screen row (A) column (Y)
F800|
F800| 4A                PLOT             LSR   A
F801| 08                                 PHP
F802| 20 47F8                            JSR   GBASCALC
F805| 28                                 PLP
F806| A9 0F                              LDA   #$0F
F808| 9002                               BCC   RTMASK
F80A| 69 E0                              ADC   #$E0
F80C| 85 2E             RTMASK           STA   MASK
F80E| B1 26             PLOT1            LDA   (GBASL),Y
F810| 45 30                              EOR   COLOR
F812| 25 2E                              AND   MASK
F814| 51 26                              EOR   (GBASL),Y
F816| 91 26                              STA   (GBASL),Y
F818| 60                                 RTS
F819|
F819|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F819|                   ; + Routine : HLINE (Horizontal LINE)             F819 [MONITOR ENTRY]
F819|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                 Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

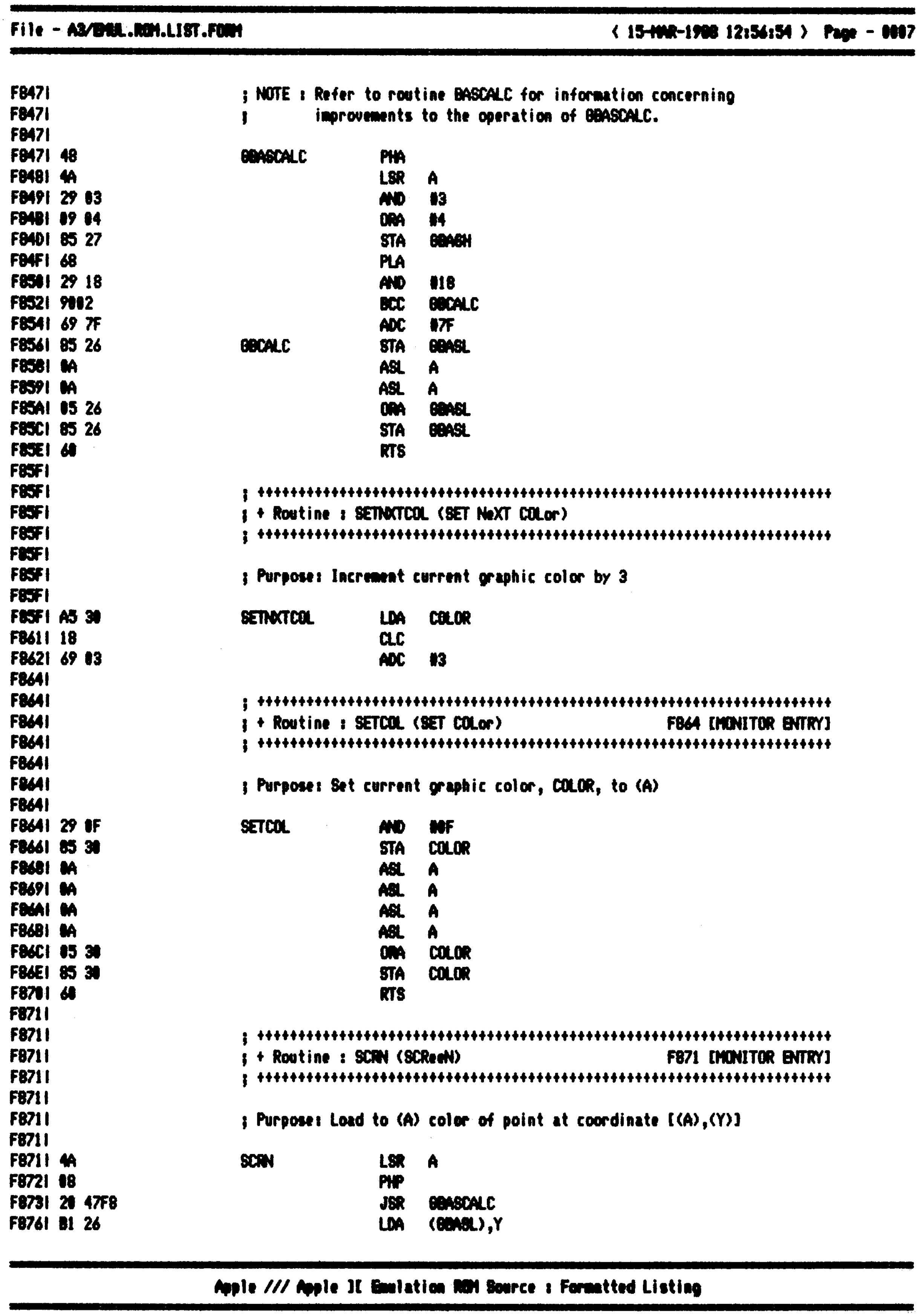


File - A3/EMUL.ROM.LIST.FORM < 15-MAR-1988 12:56:54 > Page - 0006

```
F819|
F819|                 ; Purpose: Draw horizontal lores line at row (A) from column (Y) to (H2)
F819|
F819| 20 00F8         HLINE         JSR   PLOT
F81C| C4 2C          HLINE1        CPY   H2
F81E| B011                         BCS   RTS1
F820| C8                           INY
F821| 20 0EF8                      JSR   PLOT1
F824| 90F6                         BCC   HLINE1
F826| 69 01          VLINEZ        ADC   #1
F828|
F828|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F828|                 ; + Routine : VLINE (Vertical LINE)               F828 [MONITOR ENTRY]
F828|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F828|
F828|                 ; Purpose: Draw a vertical line at column (Y) from (A) to (V2)
F828|
F828| 48             VLINE         PHA
F829| 20 00F8                      JSR   PLOT
F82C| 68                           PLA
F82D| C5 2D                        CMP   V2
F82F| 90F5                         BCC   VLINEZ
F831| 60             RTS1          RTS
F832|
F832|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F832|                 ; + Routine : CLRSCR (CLeaR SCReen)                F832 [MONITOR ENTRY]
F832|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F832|
F832|                 ; Purpose: Clear the screen (48 lines)
F832|
F832| A0 2F          CLRSCR        LDY   #2F
F834| D002                         BNE   CLRSC2
F836|
F836|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F836|                 ; + Routine : CLRTOP (CLeaR TOP)                   F836 [MONITOR ENTRY]
F836|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F836|
F836|                 ; Purpose: Clear graphics area (40 lines)
F836|
F836| A0 27          CLRTOP        LDY   #GBASH
F838| 84 2D          CLRSC2        STY   V2
F83A| A0 27                        LDY   #GBASH
F83C| A9 00          CLRSC3        LDA   #0
F83E| 85 30                        STA   COLOR
F840| 20 28F8                      JSR   VLINE
F843| 88                           DEY
F844| 10F6                         BPL   CLRSC3
F846| 60                           RTS
F847|
F847|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F847|                 ; + Routine : GBASCALC (Graphic BASe address CALCulator)
F847|                 ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F847|
F847|                 ; Purpose: Set GBASL,H from (A) (with A = line / 2)
F847|
```

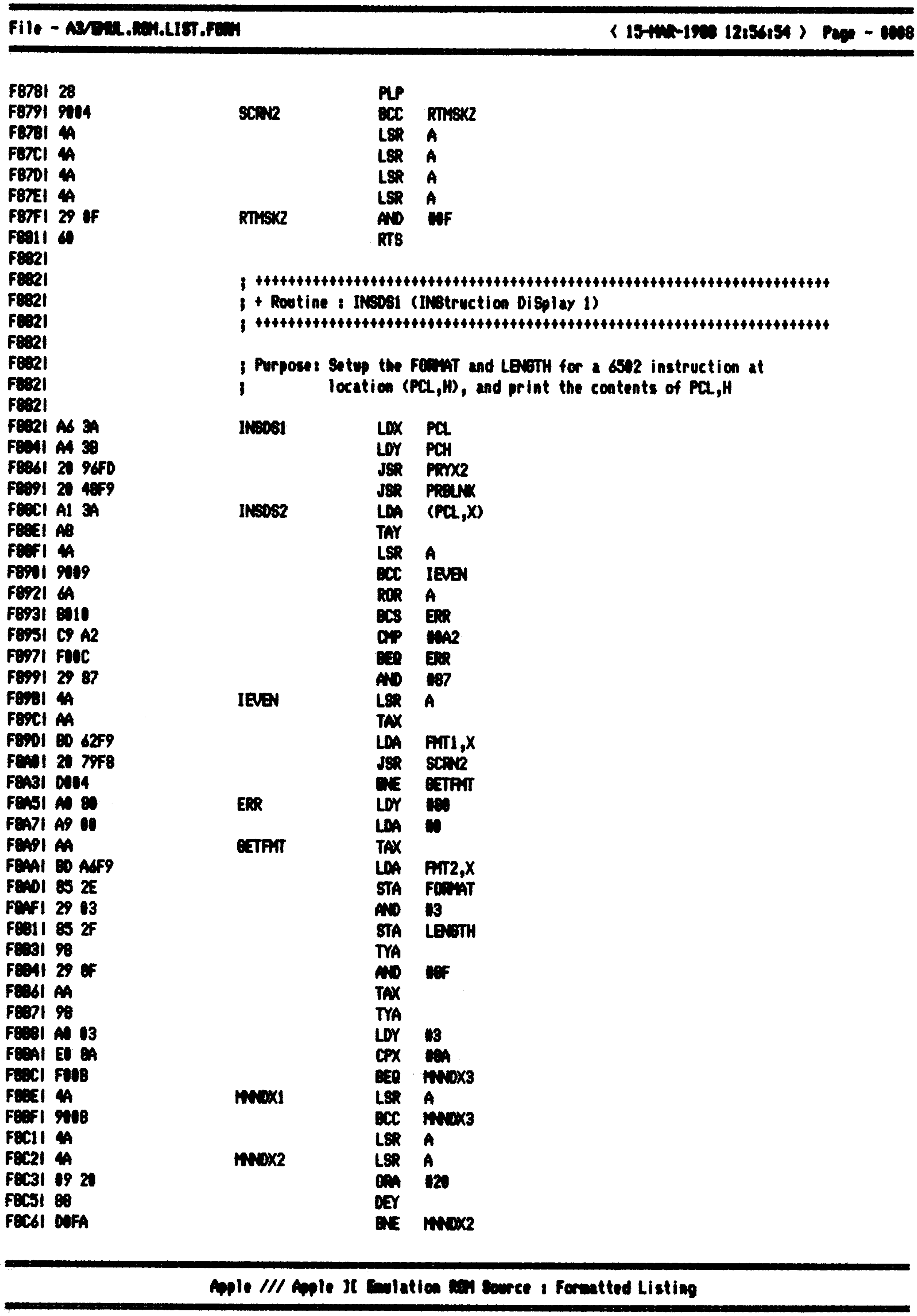
Apple /// Apple][Emulation ROM Source : Formatted Listing

"_10.PICT" 163 KB 2001-08-13 dpi: 300h x 300v pix: 2213h x 3160v

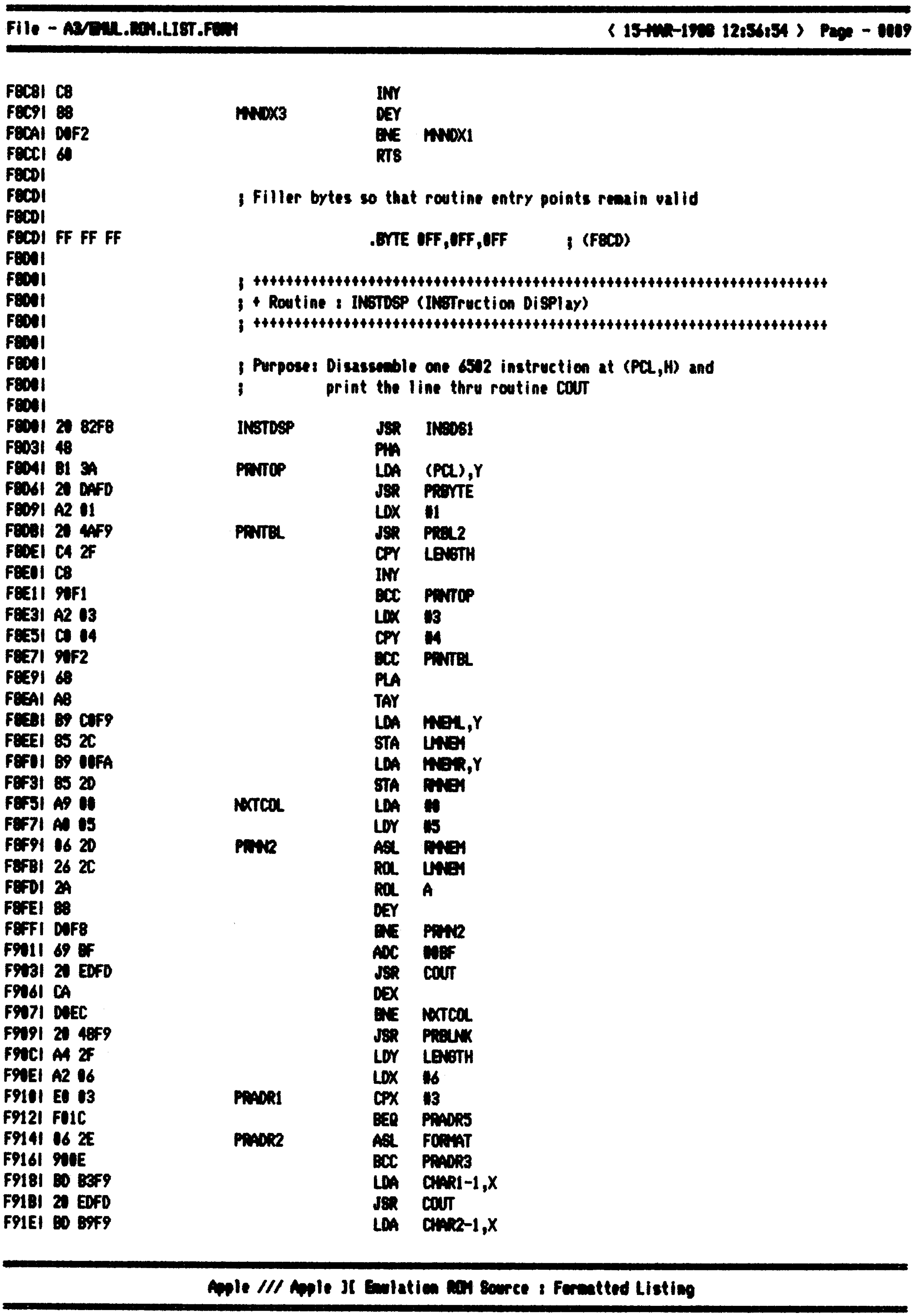


```
F847|                     ; NOTE : Refer to routine BASCALC for information concerning
F847|                     ;        improvements to the operation of GBASCALC.
F847|
F847| 48                  GBASCALC     PHA
F848| 4A                               LSR   A
F849| 29 03                            AND   #3
F84B| 09 04                            ORA   #4
F84D| 85 27                            STA   GBASH
F84F| 68                               PLA
F850| 29 18                            AND   #18
F852| 9002                             BCC   GBCALC
F854| 69 7F                            ADC   #7F
F856| 85 26               GBCALC       STA   GBASL
F858| 0A                               ASL   A
F859| 0A                               ASL   A
F85A| 05 26                            ORA   GBASL
F85C| 85 26                            STA   GBASL
F85E| 60                               RTS
F85F|
F85F|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F85F|                     ; + Routine : SETNXTCOL (SET NeXT COLor)
F85F|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F85F|
F85F|                     ; Purpose: Increment current graphic color by 3
F85F|
F85F| A5 30               SETNXTCOL    LDA   COLOR
F861| 18                               CLC
F862| 69 03                            ADC   #3
F864|
F864|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F864|                     ; + Routine : SETCOL (SET COLor)                  F864 [MONITOR ENTRY]
F864|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F864|
F864|                     ; Purpose: Set current graphic color, COLOR, to (A)
F864|
F864| 29 0F               SETCOL       AND   #0F
F866| 85 30                            STA   COLOR
F868| 0A                               ASL   A
F869| 0A                               ASL   A
F86A| 0A                               ASL   A
F86B| 0A                               ASL   A
F86C| 05 30                            ORA   COLOR
F86E| 85 30                            STA   COLOR
F870| 60                               RTS
F871|
F871|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F871|                     ; + Routine : SCRN (SCReeN)                       F871 [MONITOR ENTRY]
F871|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F871|
F871|                     ; Purpose: Load to (A) color of point at coordinate [(A),(Y)]
F871|
F871| 4A                  SCRN         LSR   A
F872| 08                               PHP
F873| 20 47F8                          JSR   GBASCALC
F876| B1 26                            LDA   (GBASL),Y
```

"_11.PICT" 151 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3178v

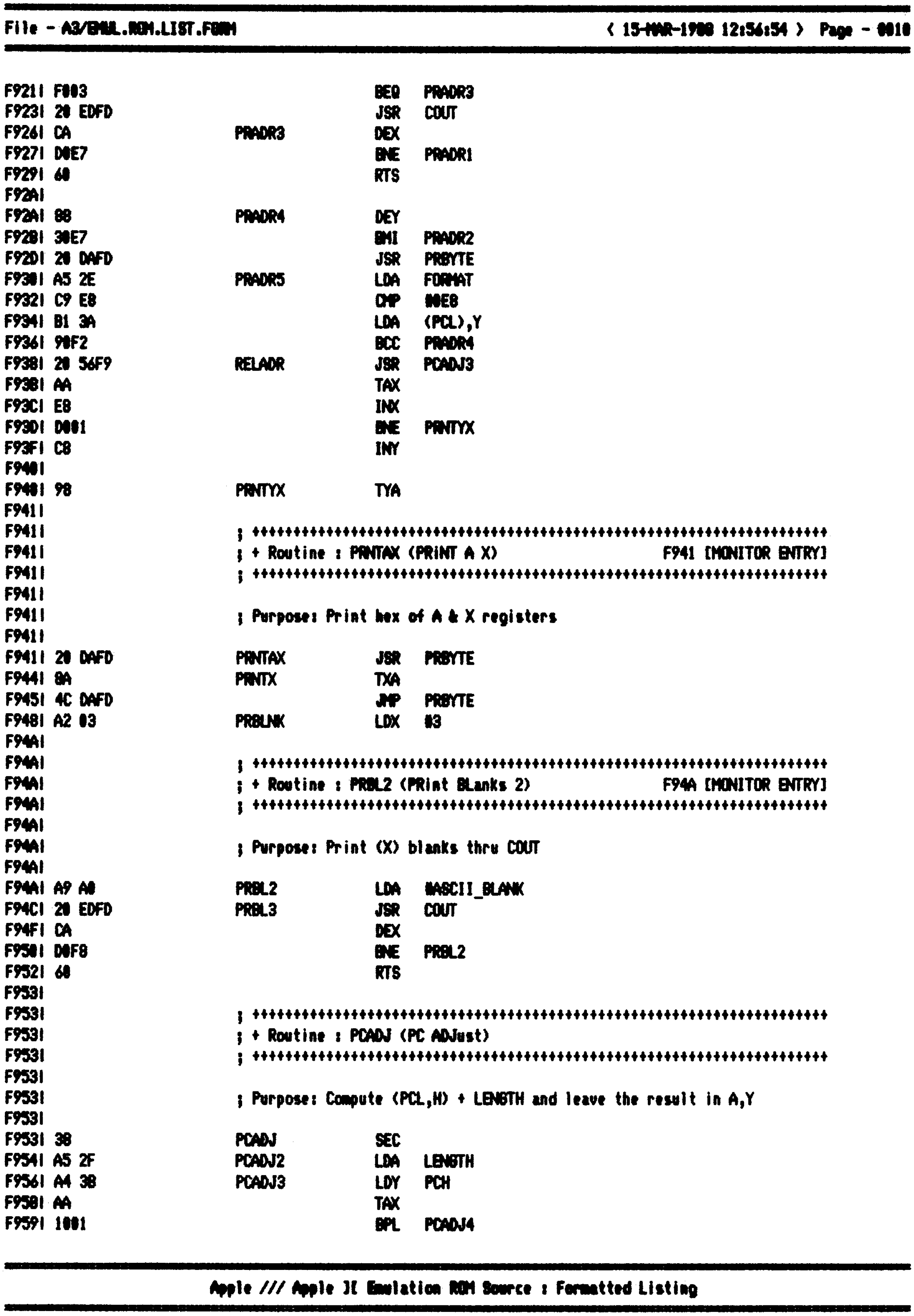


```
F878| 28                              PLP
F879| 9004          SCRN2             BCC   RTMSKZ
F87B| 4A                              LSR   A
F87C| 4A                              LSR   A
F87D| 4A                              LSR   A
F87E| 4A                              LSR   A
F87F| 29 0F         RTMSKZ            AND   #0F
F881| 60                              RTS
F882|
F882|               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F882|               ; + Routine : INSDS1 (INStruction DiSplay 1)
F882|               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F882|
F882|               ; Purpose: Setup the FORMAT and LENGTH for a 6502 instruction at
F882|               ;          location (PCL,H), and print the contents of PCL,H
F882|
F882| A6 3A         INSDS1            LDX   PCL
F884| A4 3B                           LDY   PCH
F886| 20 96FD                         JSR   PRYX2
F889| 20 48F9                         JSR   PRBLNK
F88C| A1 3A         INSDS2            LDA   (PCL,X)
F88E| A8                              TAY
F88F| 4A                              LSR   A
F890| 9009                            BCC   IEVEN
F892| 6A                              ROR   A
F893| B010                            BCS   ERR
F895| C9 A2                           CMP   #0A2
F897| F00C                            BEQ   ERR
F899| 29 87                           AND   #87
F89B| 4A            IEVEN             LSR   A
F89C| AA                              TAX
F89D| BD 62F9                         LDA   FMT1,X
F8A0| 20 79F8                         JSR   SCRN2
F8A3| D004                            BNE   GETFMT
F8A5| A0 80         ERR               LDY   #80
F8A7| A9 00                           LDA   #0
F8A9| AA            GETFMT            TAX
F8AA| BD A6F9                         LDA   FMT2,X
F8AD| 85 2E                           STA   FORMAT
F8AF| 29 03                           AND   #3
F8B1| 85 2F                           STA   LENGTH
F8B3| 98                              TYA
F8B4| 29 8F                           AND   #8F
F8B6| AA                              TAX
F8B7| 98                              TYA
F8B8| A0 03                           LDY   #3
F8BA| E0 8A                           CPX   #8A
F8BC| F00B                            BEQ   MNNDX3
F8BE| 4A            MNNDX1            LSR   A
F8BF| 9008                            BCC   MNNDX3
F8C1| 4A                              LSR   A
F8C2| 4A            MNNDX2            LSR   A
F8C3| 09 20                           ORA   #20
F8C5| 88                              DEY
F8C6| D0FA                            BNE   MNNDX2
```



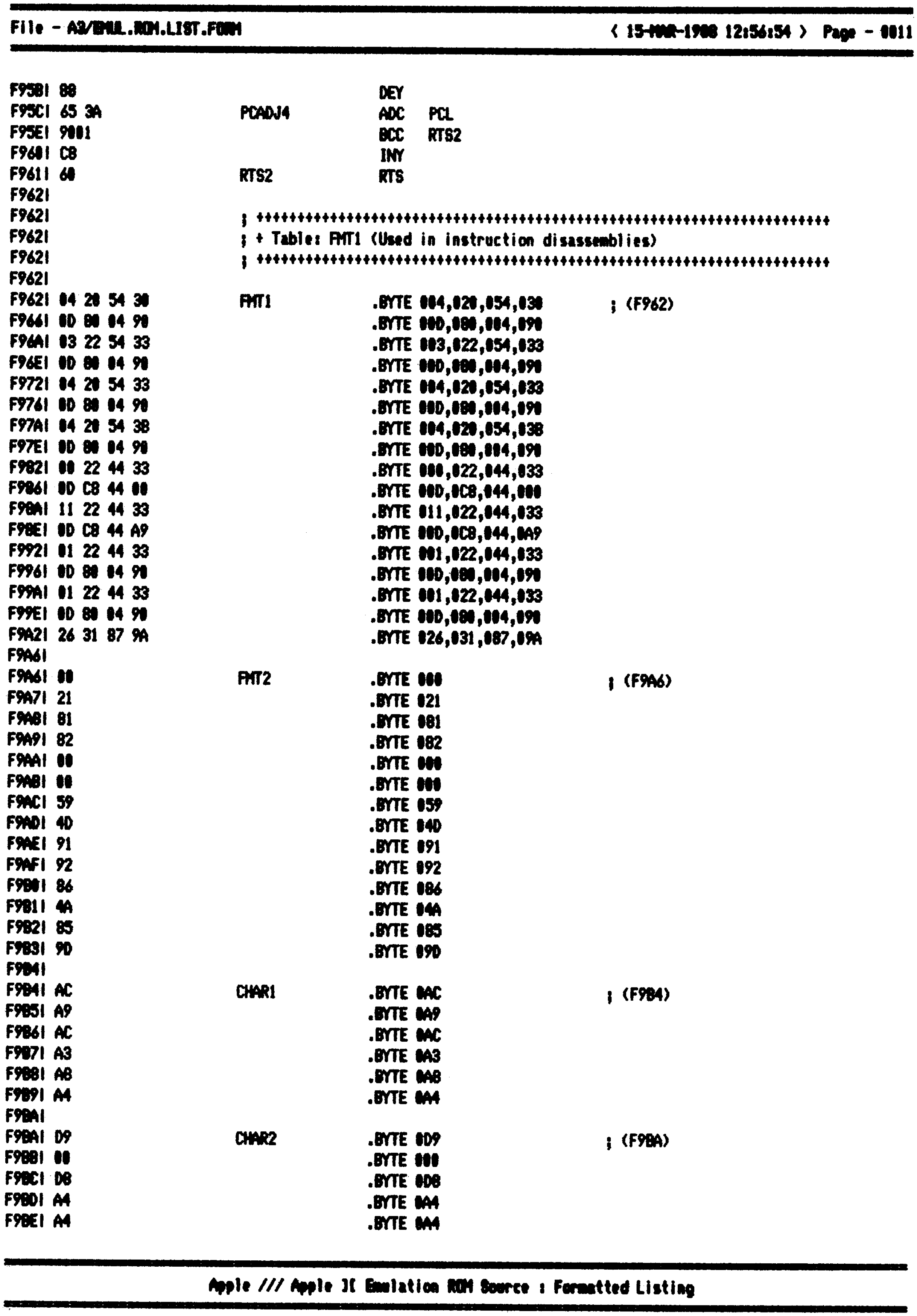
```
F8C8| C8                                  INY
F8C9| 88                 MNNDX3           DEY
F8CA| D0F2                                BNE   MNNDX1
F8CC| 60                                  RTS
F8CD|
F8CD|                    ; Filler bytes so that routine entry points remain valid
F8CD|
F8CD| FF FF FF                           .BYTE 0FF,0FF,0FF       ; (F8CD)
F8D0|
F8D0|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F8D0|                    ; + Routine : INSTDSP (INSTruction DiSPlay)
F8D0|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F8D0|
F8D0|                    ; Purpose: Disassemble one 6502 instruction at (PCL,H) and
F8D0|                    ;          print the line thru routine COUT
F8D0|
F8D0| 20 82F8            INSTDSP          JSR   INSDS1
F8D3| 48                                  PHA
F8D4| B1 3A              PRNTOP           LDA   (PCL),Y
F8D6| 20 DAFD                             JSR   PRBYTE
F8D9| A2 01                               LDX   #1
F8DB| 20 4AF9            PRNTBL           JSR   PRBL2
F8DE| C4 2F                               CPY   LENGTH
F8E0| C8                                  INY
F8E1| 90F1                                BCC   PRNTOP
F8E3| A2 03                               LDX   #3
F8E5| C0 04                               CPY   #4
F8E7| 90F2                                BCC   PRNTBL
F8E9| 68                                  PLA
F8EA| A8                                  TAY
F8EB| B9 C0F9                             LDA   MNEML,Y
F8EE| 85 2C                               STA   LMNEM
F8F0| B9 00FA                             LDA   MNEMR,Y
F8F3| 85 2D                               STA   RMNEM
F8F5| A9 00              NXTCOL           LDA   #0
F8F7| A0 05                               LDY   #5
F8F9| 06 2D              PRMN2            ASL   RMNEM
F8FB| 26 2C                               ROL   LMNEM
F8FD| 2A                                  ROL   A
F8FE| 88                                  DEY
F8FF| D0F8                                BNE   PRMN2
F901| 69 BF                               ADC   #0BF
F903| 20 EDFD                             JSR   COUT
F906| CA                                  DEX
F907| D0EC                                BNE   NXTCOL
F909| 20 48F9                             JSR   PRBLNK
F90C| A4 2F                               LDY   LENGTH
F90E| A2 06                               LDX   #6
F910| E0 03              PRADR1           CPX   #3
F912| F01C                                BEQ   PRADR5
F914| 06 2E              PRADR2           ASL   FORMAT
F916| 900E                                BCC   PRADR3
F918| BD B3F9                             LDA   CHAR1-1,X
F91B| 20 EDFD                             JSR   COUT
F91E| BD B9F9                             LDA   CHAR2-1,X
```

"_13.PICT" 143 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3172v



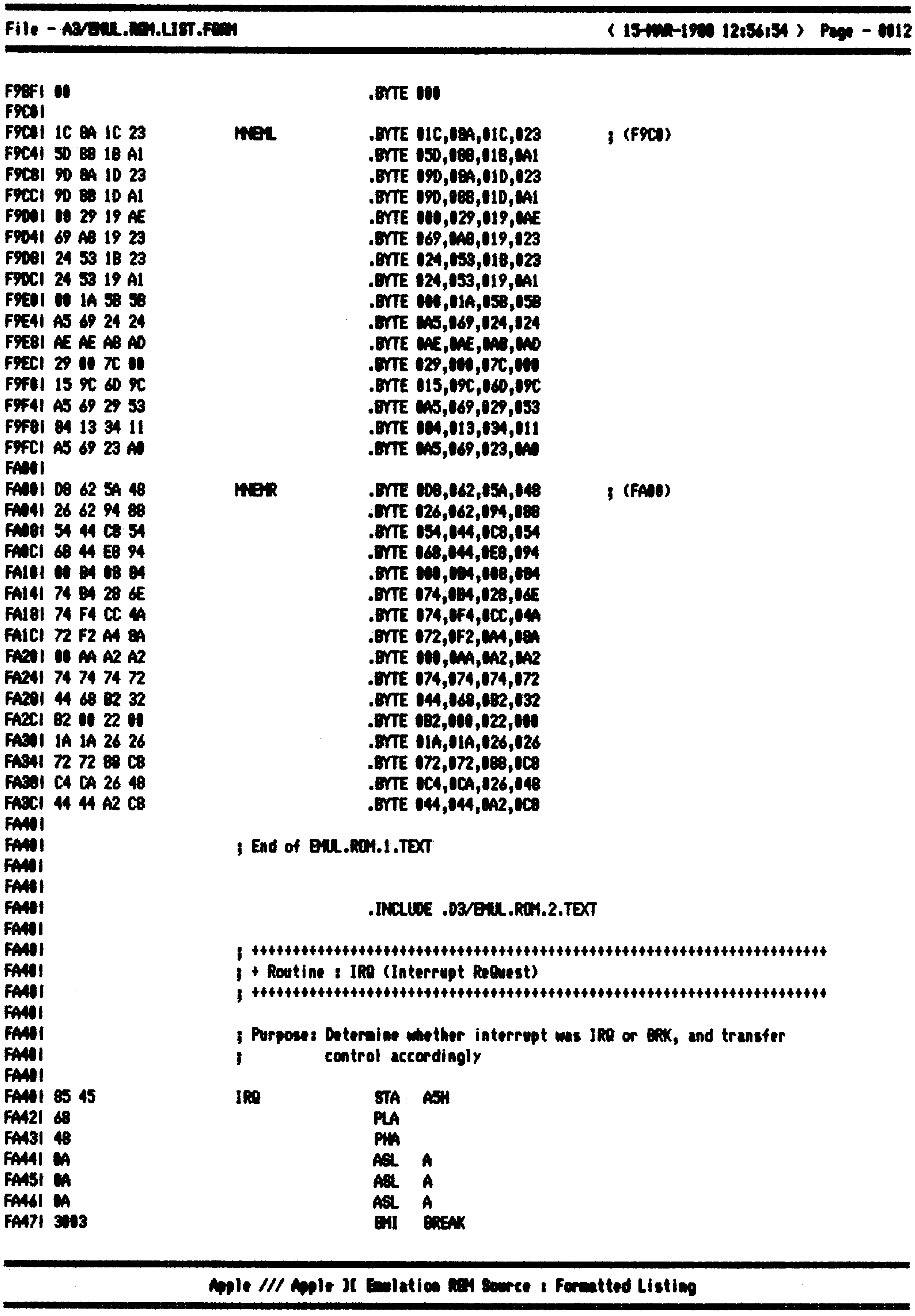
```
F921: F003                          BEQ   PRADR3
F923: 20 EDFD                       JSR   COUT
F926: CA             PRADR3         DEX
F927: D0E7                          BNE   PRADR1
F929: 60                            RTS
F92A:
F92A: 88             PRADR4         DEY
F92B: 30E7                          BMI   PRADR2
F92D: 20 DAFD                       JSR   PRBYTE
F930: A5 2E          PRADR5         LDA   FORMAT
F932: C9 E8                         CMP   #0E8
F934: B1 3A                         LDA   (PCL),Y
F936: 90F2                          BCC   PRADR4
F938: 20 56F9        RELADR         JSR   PCADJ3
F93B: AA                            TAX
F93C: E8                            INX
F93D: D001                          BNE   PRNTYX
F93F: C8                            INY
F940:
F940: 98             PRNTYX         TYA
F941:
F941:                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F941:                ; + Routine : PRNTAX (PRINT A X)                F941 [MONITOR ENTRY]
F941:                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F941:
F941:                ; Purpose: Print hex of A & X registers
F941:
F941: 20 DAFD        PRNTAX         JSR   PRBYTE
F944: 8A             PRNTX          TXA
F945: 4C DAFD                       JMP   PRBYTE
F948: A2 03          PRBLNK         LDX   #3
F94A:
F94A:                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F94A:                ; + Routine : PRBL2 (PRint BLanks 2)            F94A [MONITOR ENTRY]
F94A:                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F94A:
F94A:                ; Purpose: Print (X) blanks thru COUT
F94A:
F94A: A9 A0          PRBL2          LDA   #ASCII_BLANK
F94C: 20 EDFD        PRBL3          JSR   COUT
F94F: CA                            DEX
F950: D0F8                          BNE   PRBL2
F952: 60                            RTS
F953:
F953:                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F953:                ; + Routine : PCADJ (PC ADJust)
F953:                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F953:
F953:                ; Purpose: Compute (PCL,H) + LENGTH and leave the result in A,Y
F953:
F953: 38             PCADJ          SEC
F954: A5 2F          PCADJ2         LDA   LENGTH
F956: A4 3B          PCADJ3         LDY   PCH
F958: AA                            TAX
F959: 1001                          BPL   PCADJ4
```

"_14.PICT" 155 KB 2001-08-13 dpi: 300h x 300v pix: 2243h x 3178v



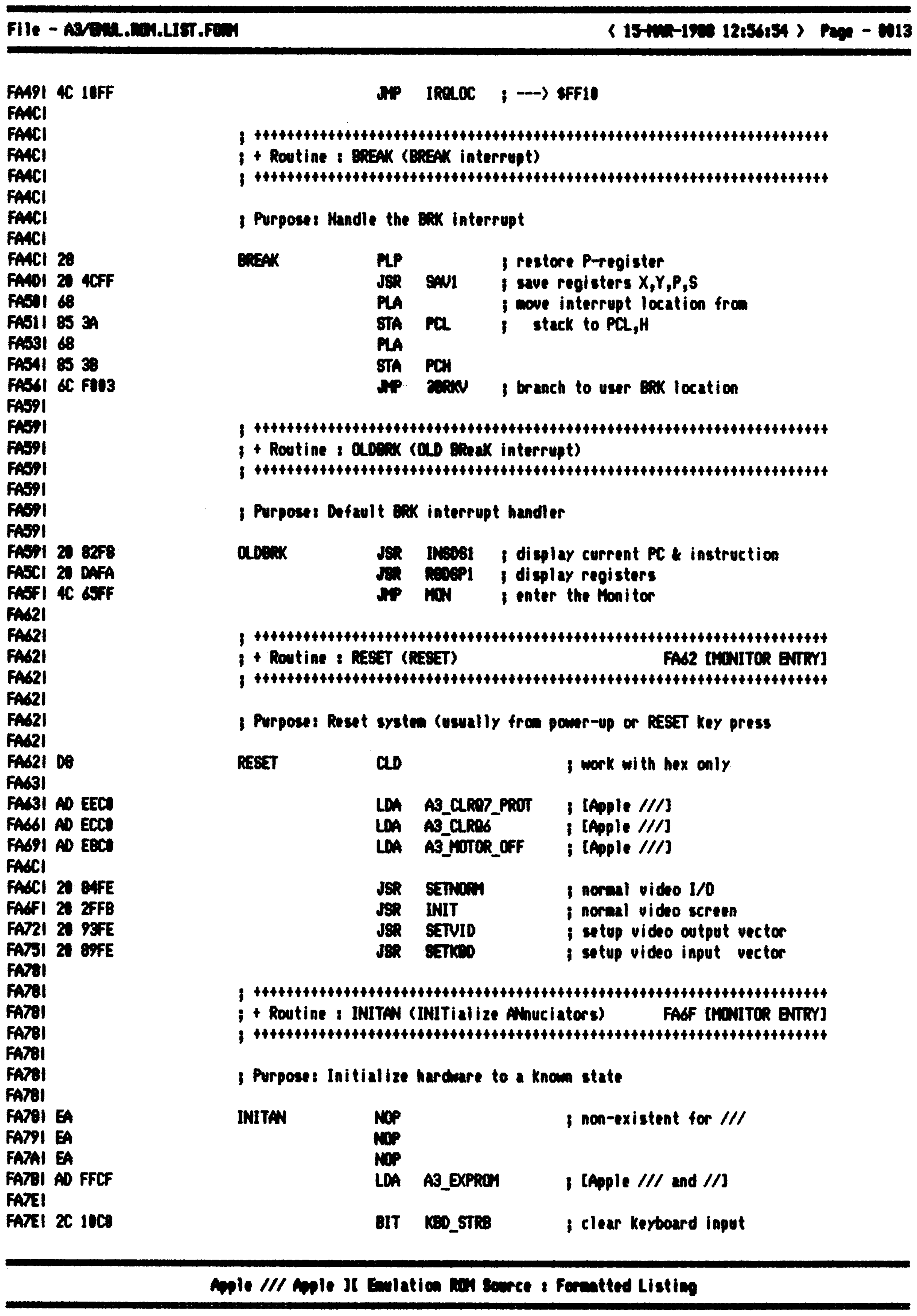
```
F95B| 88                                DEY
F95C| 65 3A              PCADJ4         ADC   PCL
F95E| 9001                              BCC   RTS2
F960| C8                                INY
F961| 60                 RTS2           RTS
F962|
F962|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F962|                    ; + Table: FMT1 (Used in instruction disassemblies)
F962|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
F962|
F962| 04 20 54 30        FMT1          .BYTE $04,$20,$54,$30         ; (F962)
F966| 0D 80 04 90                      .BYTE $0D,$80,$04,$90
F96A| 03 22 54 33                      .BYTE $03,$22,$54,$33
F96E| 0D 80 04 90                      .BYTE $0D,$80,$04,$90
F972| 04 20 54 33                      .BYTE $04,$20,$54,$33
F976| 0D 80 04 90                      .BYTE $0D,$80,$04,$90
F97A| 04 20 54 3B                      .BYTE $04,$20,$54,$3B
F97E| 0D 80 04 90                      .BYTE $0D,$80,$04,$90
F982| 00 22 44 33                      .BYTE $00,$22,$44,$33
F986| 0D C8 44 00                      .BYTE $0D,$C8,$44,$00
F98A| 11 22 44 33                      .BYTE $11,$22,$44,$33
F98E| 0D C8 44 A9                      .BYTE $0D,$C8,$44,$A9
F992| 01 22 44 33                      .BYTE $01,$22,$44,$33
F996| 0D 80 04 90                      .BYTE $0D,$80,$04,$90
F99A| 01 22 44 33                      .BYTE $01,$22,$44,$33
F99E| 0D 80 04 90                      .BYTE $0D,$80,$04,$90
F9A2| 26 31 87 9A                      .BYTE $26,$31,$87,$9A
F9A6|
F9A6| 00                 FMT2          .BYTE $00                     ; (F9A6)
F9A7| 21                               .BYTE $21
F9A8| 81                               .BYTE $81
F9A9| 82                               .BYTE $82
F9AA| 00                               .BYTE $00
F9AB| 00                               .BYTE $00
F9AC| 59                               .BYTE $59
F9AD| 4D                               .BYTE $4D
F9AE| 91                               .BYTE $91
F9AF| 92                               .BYTE $92
F9B0| 86                               .BYTE $86
F9B1| 4A                               .BYTE $4A
F9B2| 85                               .BYTE $85
F9B3| 9D                               .BYTE $9D
F9B4|
F9B4| AC                 CHAR1         .BYTE $AC                     ; (F9B4)
F9B5| A9                               .BYTE $A9
F9B6| AC                               .BYTE $AC
F9B7| A3                               .BYTE $A3
F9B8| A8                               .BYTE $A8
F9B9| A4                               .BYTE $A4
F9BA|
F9BA| D9                 CHAR2         .BYTE $D9                     ; (F9BA)
F9BB| 00                               .BYTE $00
F9BC| D8                               .BYTE $D8
F9BD| A4                               .BYTE $A4
F9BE| A4                               .BYTE $A4
```

"_15.PICT" 155 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3172v



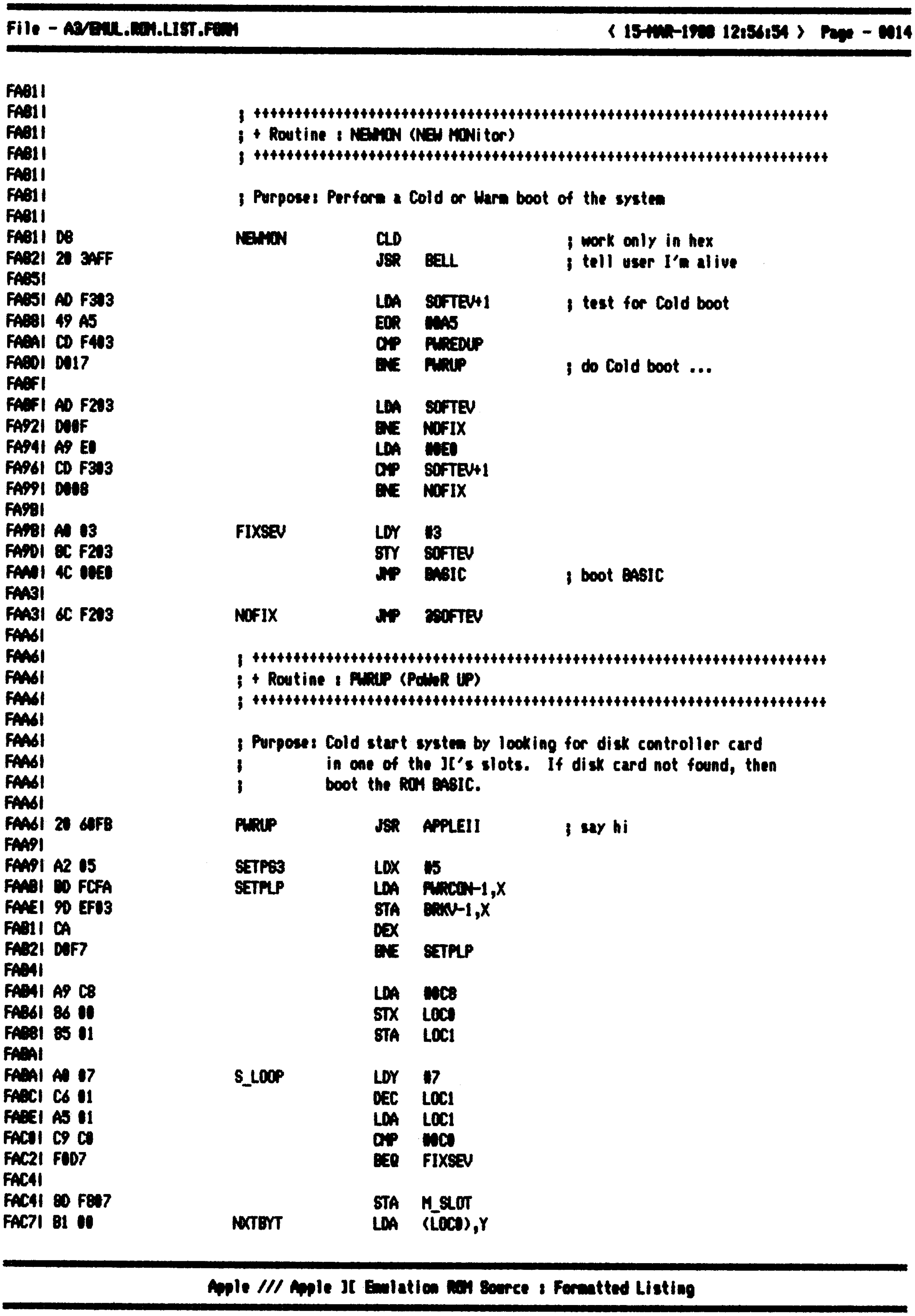
```
F9BF| 00                            .BYTE 000
F9C0|
F9C0| 1C 8A 1C 23       MNEML       .BYTE 01C,08A,01C,023       ; (F9C0)
F9C4| 5D 8B 1B A1                   .BYTE 05D,08B,01B,0A1
F9C8| 9D 8A 1D 23                   .BYTE 09D,08A,01D,023
F9CC| 9D 8B 1D A1                   .BYTE 09D,08B,01D,0A1
F9D0| 00 29 19 AE                   .BYTE 000,029,019,0AE
F9D4| 69 A8 19 23                   .BYTE 069,0A8,019,023
F9D8| 24 53 1B 23                   .BYTE 024,053,01B,023
F9DC| 24 53 19 A1                   .BYTE 024,053,019,0A1
F9E0| 00 1A 5B 5B                   .BYTE 000,01A,05B,05B
F9E4| A5 69 24 24                   .BYTE 0A5,069,024,024
F9E8| AE AE A8 AD                   .BYTE 0AE,0AE,0A8,0AD
F9EC| 29 00 7C 00                   .BYTE 029,000,07C,000
F9F0| 15 9C 6D 9C                   .BYTE 015,09C,06D,09C
F9F4| A5 69 29 53                   .BYTE 0A5,069,029,053
F9F8| 84 13 34 11                   .BYTE 084,013,034,011
F9FC| A5 69 23 A0                   .BYTE 0A5,069,023,0A0
FA00|
FA00| D8 62 5A 48       MNEMR       .BYTE 0D8,062,05A,048       ; (FA00)
FA04| 26 62 94 88                   .BYTE 026,062,094,088
FA08| 54 44 C8 54                   .BYTE 054,044,0C8,054
FA0C| 68 44 E8 94                   .BYTE 068,044,0E8,094
FA10| 00 B4 08 84                   .BYTE 000,0B4,008,084
FA14| 74 B4 28 6E                   .BYTE 074,0B4,028,06E
FA18| 74 F4 CC 4A                   .BYTE 074,0F4,0CC,04A
FA1C| 72 F2 A4 8A                   .BYTE 072,0F2,0A4,08A
FA20| 00 AA A2 A2                   .BYTE 000,0AA,0A2,0A2
FA24| 74 74 74 72                   .BYTE 074,074,074,072
FA28| 44 68 B2 32                   .BYTE 044,068,0B2,032
FA2C| B2 00 22 00                   .BYTE 0B2,000,022,000
FA30| 1A 1A 26 26                   .BYTE 01A,01A,026,026
FA34| 72 72 88 C8                   .BYTE 072,072,088,0C8
FA38| C4 CA 26 48                   .BYTE 0C4,0CA,026,048
FA3C| 44 44 A2 C8                   .BYTE 044,044,0A2,0C8
FA40|
FA40|                   ; End of EMUL.ROM.1.TEXT
FA40|
FA40|
FA40|                               .INCLUDE .D3/EMUL.ROM.2.TEXT
FA40|
FA40|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA40|                   ; + Routine : IRQ (Interrupt ReQuest)
FA40|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA40|
FA40|                   ; Purpose: Determine whether interrupt was IRQ or BRK, and transfer
FA40|                   ;          control accordingly
FA40|
FA40| 85 45             IRQ          STA   A5H
FA42| 68                             PLA
FA43| 48                             PHA
FA44| 0A                             ASL   A
FA45| 0A                             ASL   A
FA46| 0A                             ASL   A
FA47| 3003                           BMI   BREAK
```

“_16.PICT” 180 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3166v

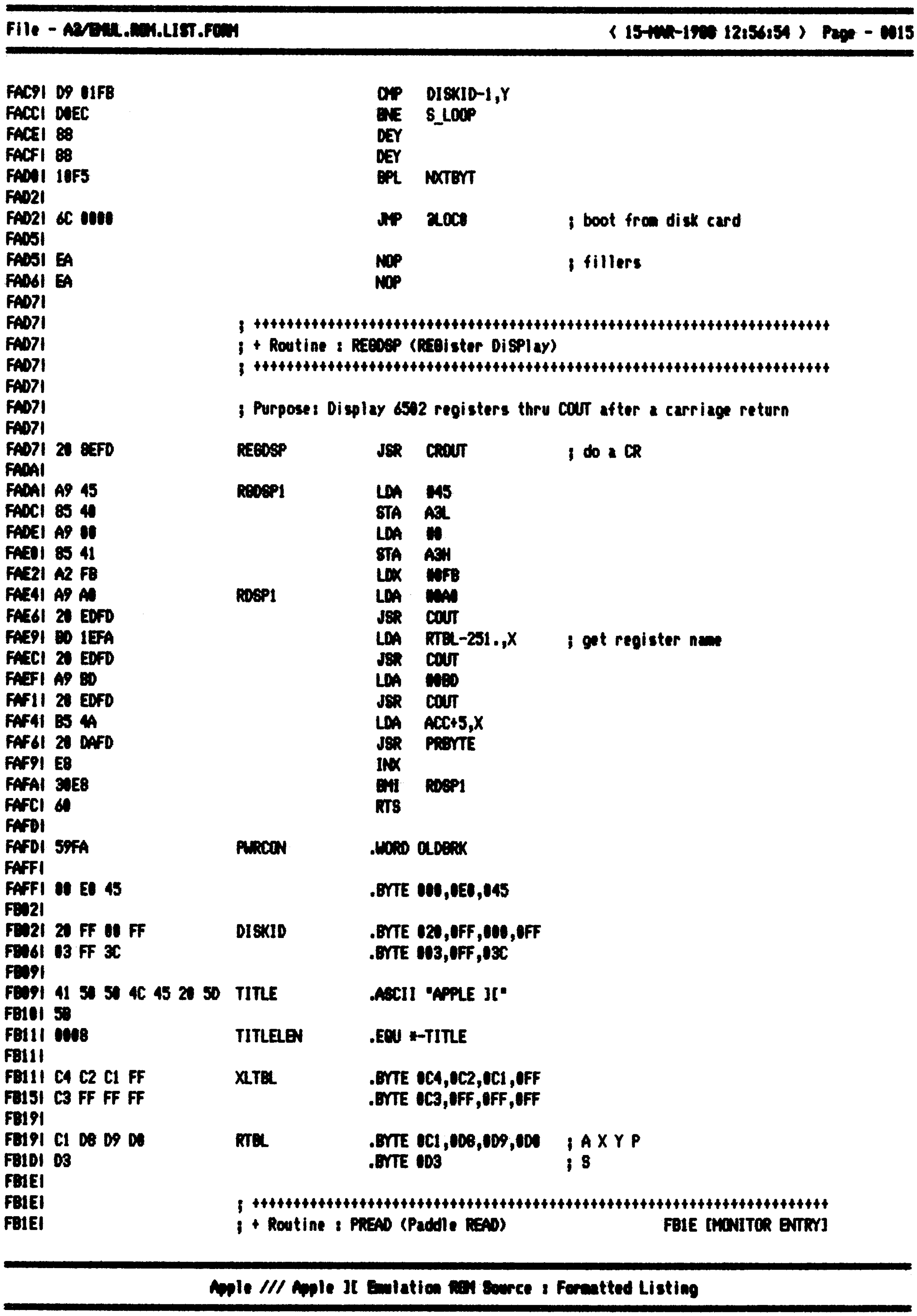


```
File - A3/EMUL.ROM.LIST.FORM                                    ( 15-MAR-1988 12:56:54 )  Page - 0013

FA49| 4C 10FF                       JMP   IRQLOC   ; ---> $FF10
FA4C|
FA4C|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA4C|                  ; + Routine : BREAK (BREAK interrupt)
FA4C|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA4C|
FA4C|                  ; Purpose: Handle the BRK interrupt
FA4C|
FA4C| 28               BREAK        PLP            ; restore P-register
FA4D| 20 4CFF                       JSR   SAV1     ; save registers X,Y,P,S
FA50| 68                            PLA            ; move interrupt location from
FA51| 85 3A                         STA   PCL      ;   stack to PCL,H
FA53| 68                            PLA
FA54| 85 3B                         STA   PCH
FA56| 6C F003                       JMP   (BRKV    ; branch to user BRK location
FA59|
FA59|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA59|                  ; + Routine : OLDBRK (OLD BReaK interrupt)
FA59|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA59|
FA59|                  ; Purpose: Default BRK interrupt handler
FA59|
FA59| 20 82F8          OLDBRK       JSR   INSDS1   ; display current PC & instruction
FA5C| 20 DAFA                       JSR   RGDSP1   ; display registers
FA5F| 4C 65FF                       JMP   MON      ; enter the Monitor
FA62|
FA62|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA62|                  ; + Routine : RESET (RESET)                          FA62 [MONITOR ENTRY]
FA62|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA62|
FA62|                  ; Purpose: Reset system (usually from power-up or RESET key press
FA62|
FA62| D8               RESET        CLD                     ; work with hex only
FA63|
FA63| AD EEC0                      LDA   A3_CLRQ7_PROT     ; [Apple ///]
FA66| AD ECC0                      LDA   A3_CLRQ6          ; [Apple ///]
FA69| AD E8C0                      LDA   A3_MOTOR_OFF      ; [Apple ///]
FA6C|
FA6C| 20 84FE                       JSR   SETNORM           ; normal video I/O
FA6F| 20 2FFB                       JSR   INIT              ; normal video screen
FA72| 20 93FE                       JSR   SETVID            ; setup video output vector
FA75| 20 89FE                       JSR   SETKBD            ; setup video input  vector
FA78|
FA78|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA78|                  ; + Routine : INITAN (INITialize ANnuciators)        FA6F [MONITOR ENTRY]
FA78|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA78|
FA78|                  ; Purpose: Initialize hardware to a known state
FA78|
FA78| EA               INITAN       NOP                     ; non-existent for ///
FA79| EA                            NOP
FA7A| EA                            NOP
FA7B| AD FFCF                       LDA   A3_EXPROM         ; [Apple /// and //]
FA7E|
FA7E| 2C 10C0                       BIT   KBD_STRB          ; clear keyboard input

                    Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

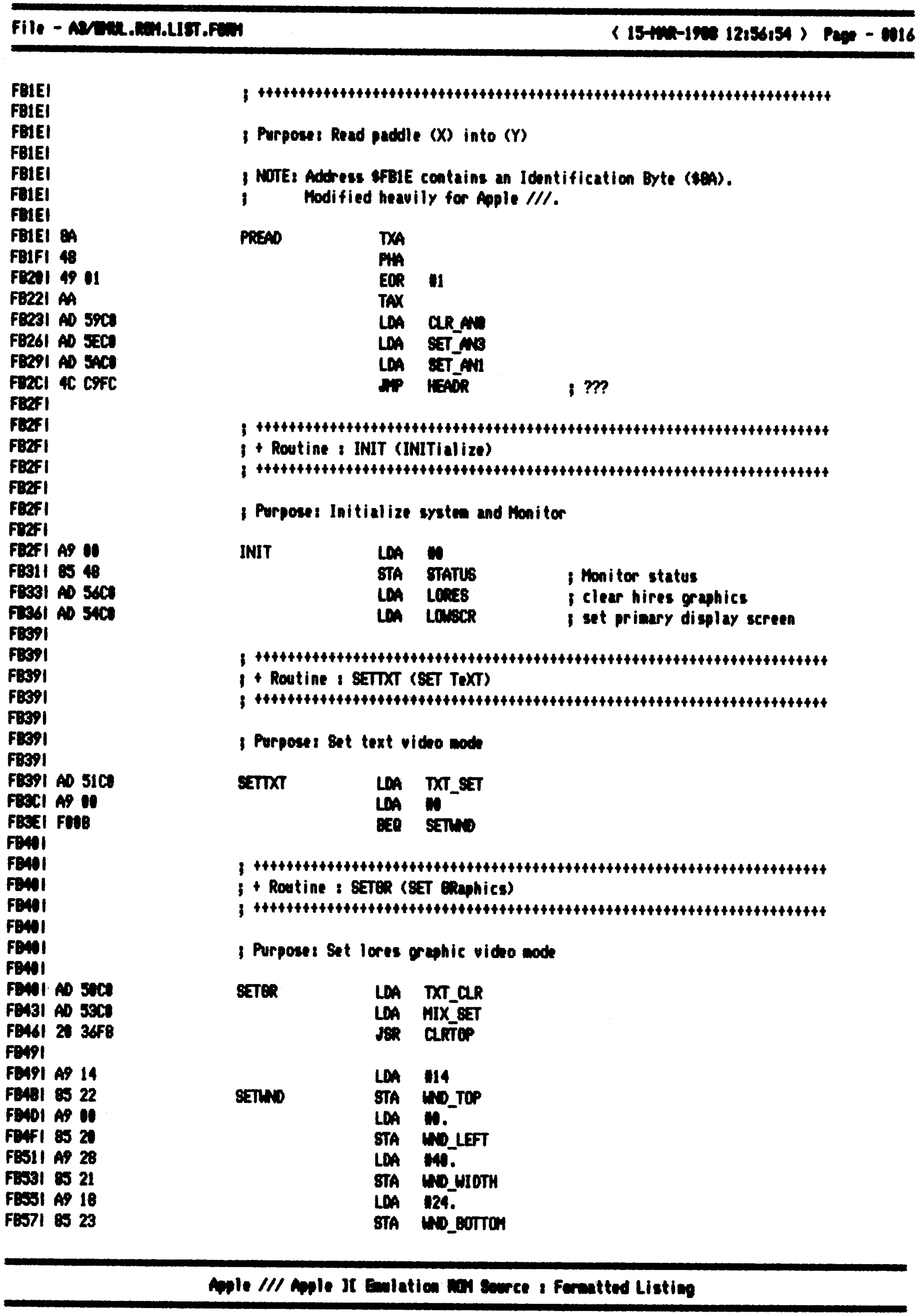
"_17.PICT" 187 KB 2001-08-13 dpi: 300h x 300v pix: 2237h x 3172v



```
FA81|
FA81|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA81|                     ; + Routine : NEWMON (NEW MONitor)
FA81|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FA81|
FA81|                     ; Purpose: Perform a Cold or Warm boot of the system
FA81|
FA81| D8                  NEWMON      CLD                     ; work only in hex
FA82| 20 3AFF                         JSR   BELL              ; tell user I'm alive
FA85|
FA85| AD F303                         LDA   SOFTEV+1          ; test for Cold boot
FA88| 49 A5                           EOR   #0A5
FA8A| CD F403                         CMP   PWREDUP
FA8D| D017                            BNE   PWRUP             ; do Cold boot ...
FA8F|
FA8F| AD F203                         LDA   SOFTEV
FA92| D00F                            BNE   NOFIX
FA94| A9 E0                           LDA   #0E0
FA96| CD F303                         CMP   SOFTEV+1
FA99| D008                            BNE   NOFIX
FA9B|
FA9B| A0 03               FIXSEV      LDY   #3
FA9D| 8C F203                         STY   SOFTEV
FAA0| 4C 00E0                         JMP   BASIC             ; boot BASIC
FAA3|
FAA3| 6C F203             NOFIX       JMP   @SOFTEV
FAA6|
FAA6|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FAA6|                     ; + Routine : PWRUP (PoWeR UP)
FAA6|                     ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FAA6|
FAA6|                     ; Purpose: Cold start system by looking for disk controller card
FAA6|                     ;          in one of the ]['s slots.  If disk card not found, then
FAA6|                     ;          boot the ROM BASIC.
FAA6|
FAA6| 20 60FB             PWRUP       JSR   APPLEII           ; say hi
FAA9|
FAA9| A2 05               SETPG3      LDX   #5
FAAB| BD FCFA             SETPLP      LDA   PWRCON-1,X
FAAE| 9D EF03                         STA   BRKV-1,X
FAB1| CA                              DEX
FAB2| D0F7                            BNE   SETPLP
FAB4|
FAB4| A9 C8                           LDA   #0C8
FAB6| 86 00                           STX   LOC0
FAB8| 85 01                           STA   LOC1
FABA|
FABA| A0 07               S_LOOP      LDY   #7
FABC| C6 01                           DEC   LOC1
FABE| A5 01                           LDA   LOC1
FAC0| C9 C0                           CMP   #0C0
FAC2| F0D7                            BEQ   FIXSEV
FAC4|
FAC4| 8D F807                         STA   M_SLOT
FAC7| B1 00               NXTBYT      LDA   (LOC0),Y
```

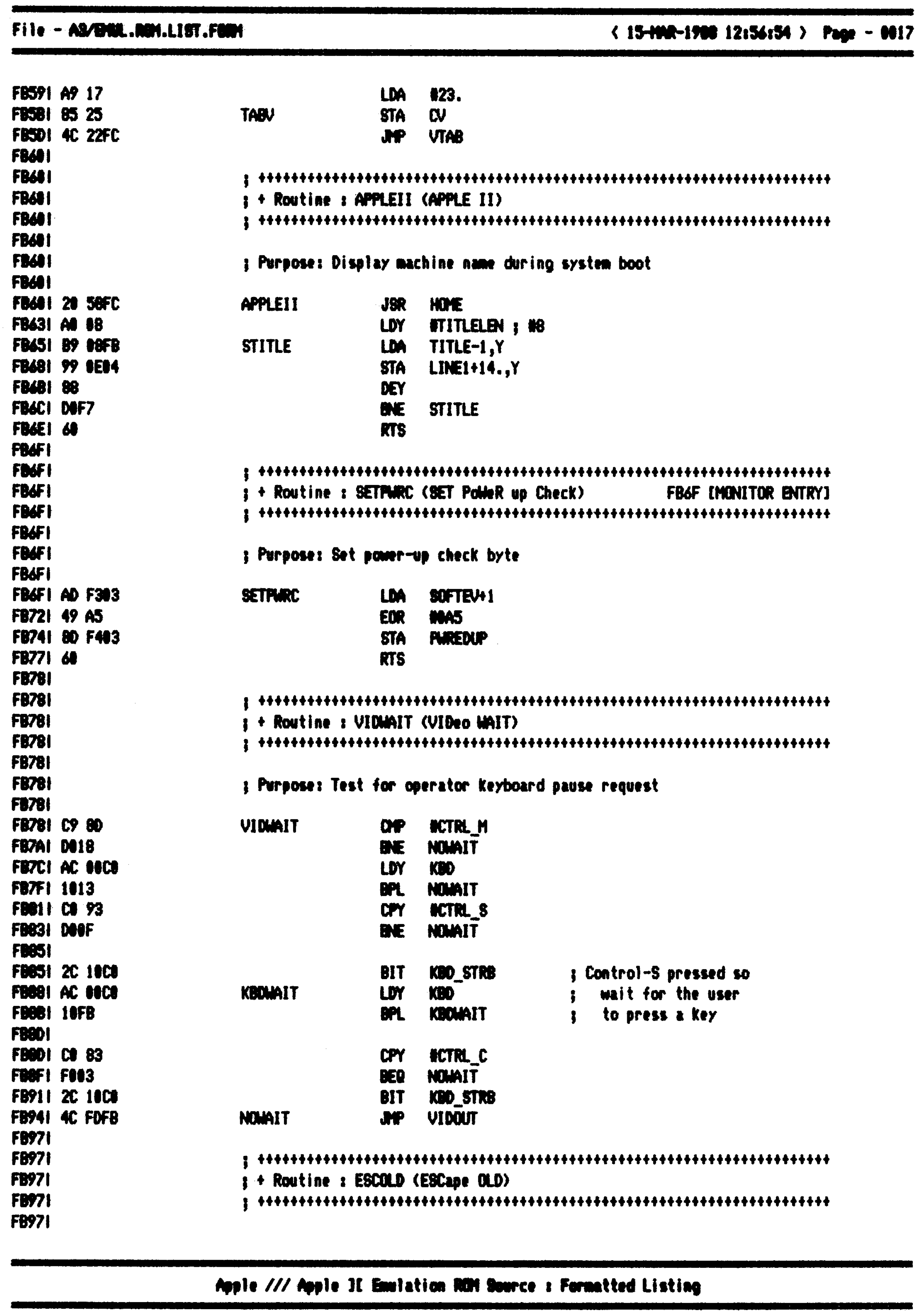


```
File - A2/EMUL.ROM.LIST.FORM                                    ( 15-MAR-1980 12:56:54 )  Page - 0015

FAC9| D9 01FB                      CMP   DISKID-1,Y
FACC| D0EC                         BNE   S_LOOP
FACE| 88                           DEY
FACF| 88                           DEY
FAD0| 10F5                         BPL   NXTBYT
FAD2|
FAD2| 6C 0000                      JMP   2LOC0            ; boot from disk card
FAD5|
FAD5| EA                           NOP                    ; fillers
FAD6| EA                           NOP
FAD7|
FAD7|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FAD7|                      ; + Routine : REGDSP (REGister DiSPlay)
FAD7|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FAD7|
FAD7|                      ; Purpose: Display 6502 registers thru COUT after a carriage return
FAD7|
FAD7| 20 8EFD              REGDSP       JSR   CROUT            ; do a CR
FADA|
FADA| A9 45                RGDSP1       LDA   #45
FADC| 85 40                             STA   A3L
FADE| A9 00                             LDA   #0
FAE0| 85 41                             STA   A3H
FAE2| A2 FB                             LDX   #0FB
FAE4| A9 A0                RDSP1        LDA   #0A0
FAE6| 20 EDFD                           JSR   COUT
FAE9| BD 1EFA                           LDA   RTBL-251.,X      ; get register name
FAEC| 20 EDFD                           JSR   COUT
FAEF| A9 BD                             LDA   #0BD
FAF1| 20 EDFD                           JSR   COUT
FAF4| B5 4A                             LDA   ACC+5,X
FAF6| 20 DAFD                           JSR   PRBYTE
FAF9| E8                                INX
FAFA| 30E8                              BMI   RDSP1
FAFC| 60                                RTS
FAFD|
FAFD| 59FA                 PWRCON       .WORD OLDBRK
FAFF|
FAFF| 00 E0 45                          .BYTE 000,0E0,045
FB02|
FB02| 20 FF 00 FF          DISKID       .BYTE 020,0FF,000,0FF
FB06| 03 FF 3C                          .BYTE 003,0FF,03C
FB09|
FB09| 41 50 50 4C 45 20 5D TITLE        .ASCII "APPLE ]["
FB10| 5B
FB11| 0008                 TITLELEN     .EQU *-TITLE
FB11|
FB11| C4 C2 C1 FF          XLTBL        .BYTE 0C4,0C2,0C1,0FF
FB15| C3 FF FF FF                       .BYTE 0C3,0FF,0FF,0FF
FB19|
FB19| C1 D8 D9 D0          RTBL         .BYTE 0C1,0D8,0D9,0D0   ; A X Y P
FB1D| D3                                .BYTE 0D3               ; S
FB1E|
FB1E|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB1E|                      ; + Routine : PREAD (Paddle READ)                     FB1E [MONITOR ENTRY]

                        Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

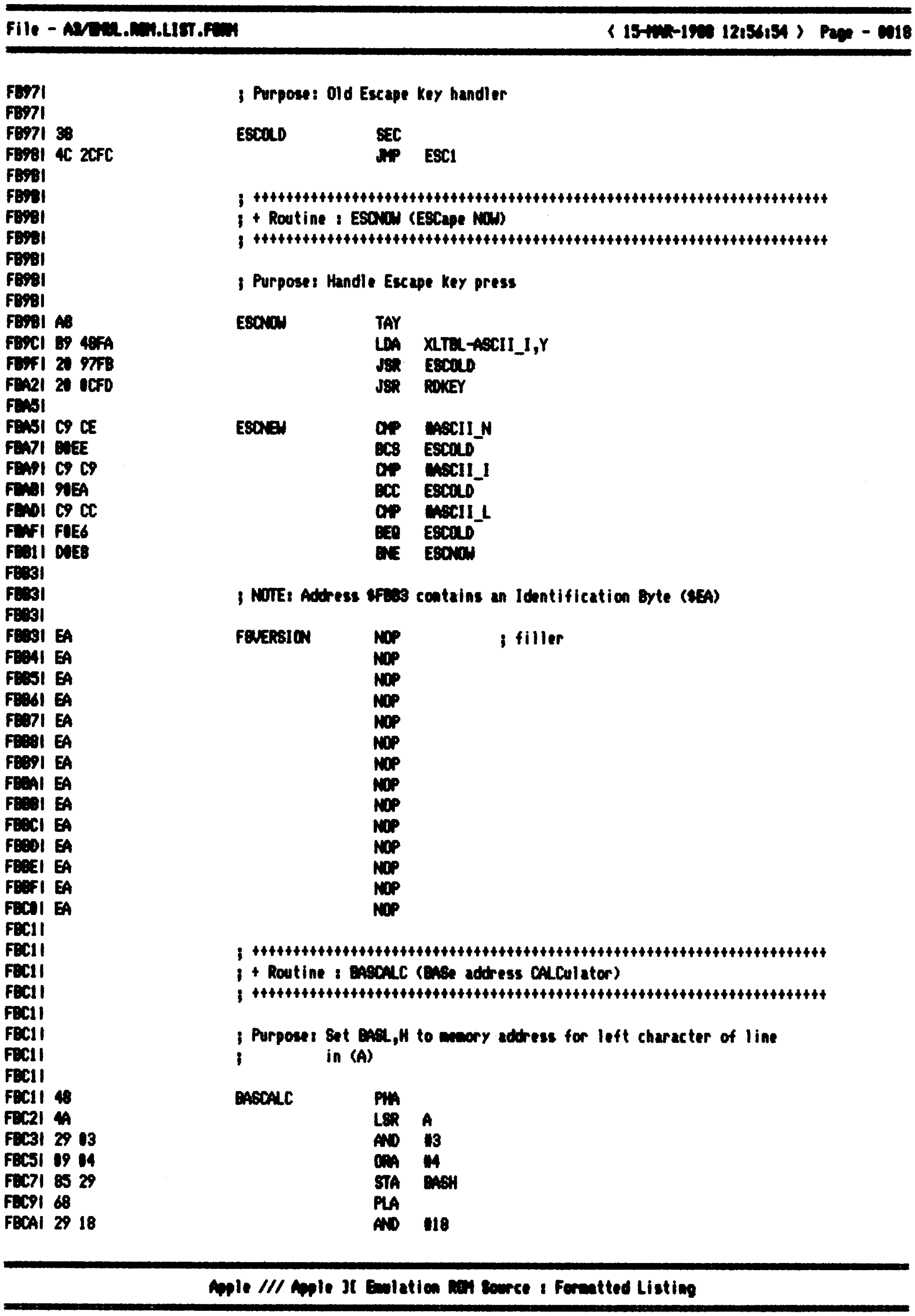


```
FB1E|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB1E|
FB1E|                   ; Purpose: Read paddle (X) into (Y)
FB1E|
FB1E|                   ; NOTE: Address $FB1E contains an Identification Byte ($8A).
FB1E|                   ;       Modified heavily for Apple ///.
FB1E|
FB1E| 8A                PREAD       TXA
FB1F| 48                            PHA
FB20| 49 01                         EOR   #1
FB22| AA                            TAX
FB23| AD 59C0                       LDA   CLR_AN0
FB26| AD 5EC0                       LDA   SET_AN3
FB29| AD 5AC0                       LDA   SET_AN1
FB2C| 4C C9FC                       JMP   HEADR           ; ???
FB2F|
FB2F|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB2F|                   ; + Routine : INIT (INITialize)
FB2F|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB2F|
FB2F|                   ; Purpose: Initialize system and Monitor
FB2F|
FB2F| A9 00             INIT        LDA   #0
FB31| 85 48                         STA   STATUS          ; Monitor status
FB33| AD 56C0                       LDA   LORES           ; clear hires graphics
FB36| AD 54C0                       LDA   LOWSCR          ; set primary display screen
FB39|
FB39|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB39|                   ; + Routine : SETTXT (SET TeXT)
FB39|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB39|
FB39|                   ; Purpose: Set text video mode
FB39|
FB39| AD 51C0           SETTXT      LDA   TXT_SET
FB3C| A9 00                         LDA   #0
FB3E| F00B                          BEQ   SETWND
FB40|
FB40|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB40|                   ; + Routine : SETGR (SET GRaphics)
FB40|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB40|
FB40|                   ; Purpose: Set lores graphic video mode
FB40|
FB40| AD 50C0           SETGR       LDA   TXT_CLR
FB43| AD 53C0                       LDA   MIX_SET
FB46| 20 36F8                       JSR   CLRTOP
FB49|
FB49| A9 14                         LDA   #14
FB4B| 85 22             SETWND      STA   WND_TOP
FB4D| A9 00                         LDA   #0.
FB4F| 85 20                         STA   WND_LEFT
FB51| A9 28                         LDA   #40.
FB53| 85 21                         STA   WND_WIDTH
FB55| A9 18                         LDA   #24.
FB57| 85 23                         STA   WND_BOTTOM
```

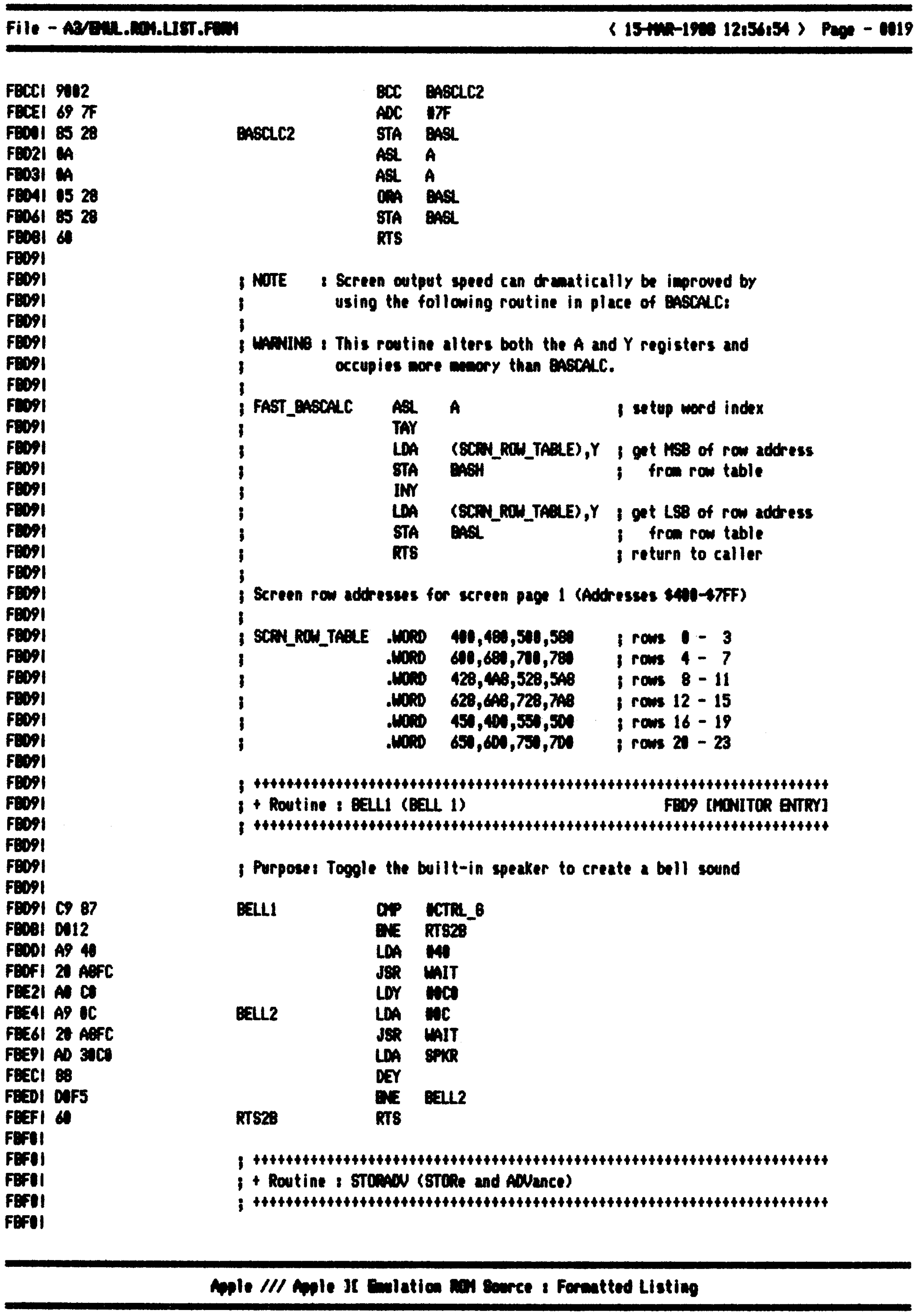
"_20.PICT" 160 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3172v



```
FB59| A9 17                           LDA   #23.
FB5B| 85 25           TABV            STA   CV
FB5D| 4C 22FC                         JMP   VTAB
FB60|
FB60|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB60|                 ; + Routine : APPLEII (APPLE II)
FB60|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB60|
FB60|                 ; Purpose: Display machine name during system boot
FB60|
FB60| 20 58FC        APPLEII         JSR   HOME
FB63| A0 08                           LDY   #TITLELEN ; #8
FB65| B9 08FB        STITLE          LDA   TITLE-1,Y
FB68| 99 0E04                         STA   LINE1+14.,Y
FB6B| 88                              DEY
FB6C| D0F7                            BNE   STITLE
FB6E| 60                              RTS
FB6F|
FB6F|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB6F|                 ; + Routine : SETPWRC (SET PoWeR up Check)          FB6F [MONITOR ENTRY]
FB6F|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB6F|
FB6F|                 ; Purpose: Set power-up check byte
FB6F|
FB6F| AD F303        SETPWRC         LDA   SOFTEV+1
FB72| 49 A5                           EOR   #0A5
FB74| 8D F403                         STA   PWREDUP
FB77| 60                              RTS
FB78|
FB78|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB78|                 ; + Routine : VIDWAIT (VIDeo WAIT)
FB78|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB78|
FB78|                 ; Purpose: Test for operator keyboard pause request
FB78|
FB78| C9 8D          VIDWAIT         CMP   #CTRL_M
FB7A| D018                            BNE   NOWAIT
FB7C| AC 00C0                         LDY   KBD
FB7F| 1013                            BPL   NOWAIT
FB81| C0 93                           CPY   #CTRL_S
FB83| D00F                            BNE   NOWAIT
FB85|
FB85| 2C 10C0                         BIT   KBD_STRB      ; Control-S pressed so
FB88| AC 00C0        KBDWAIT         LDY   KBD           ;   wait for the user
FB8B| 10FB                            BPL   KBDWAIT       ;    to press a key
FB8D|
FB8D| C0 83                           CPY   #CTRL_C
FB8F| F003                            BEQ   NOWAIT
FB91| 2C 10C0                         BIT   KBD_STRB
FB94| 4C FDFB        NOWAIT          JMP   VIDOUT
FB97|
FB97|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB97|                 ; + Routine : ESCOLD (ESCape OLD)
FB97|                 ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB97|
```

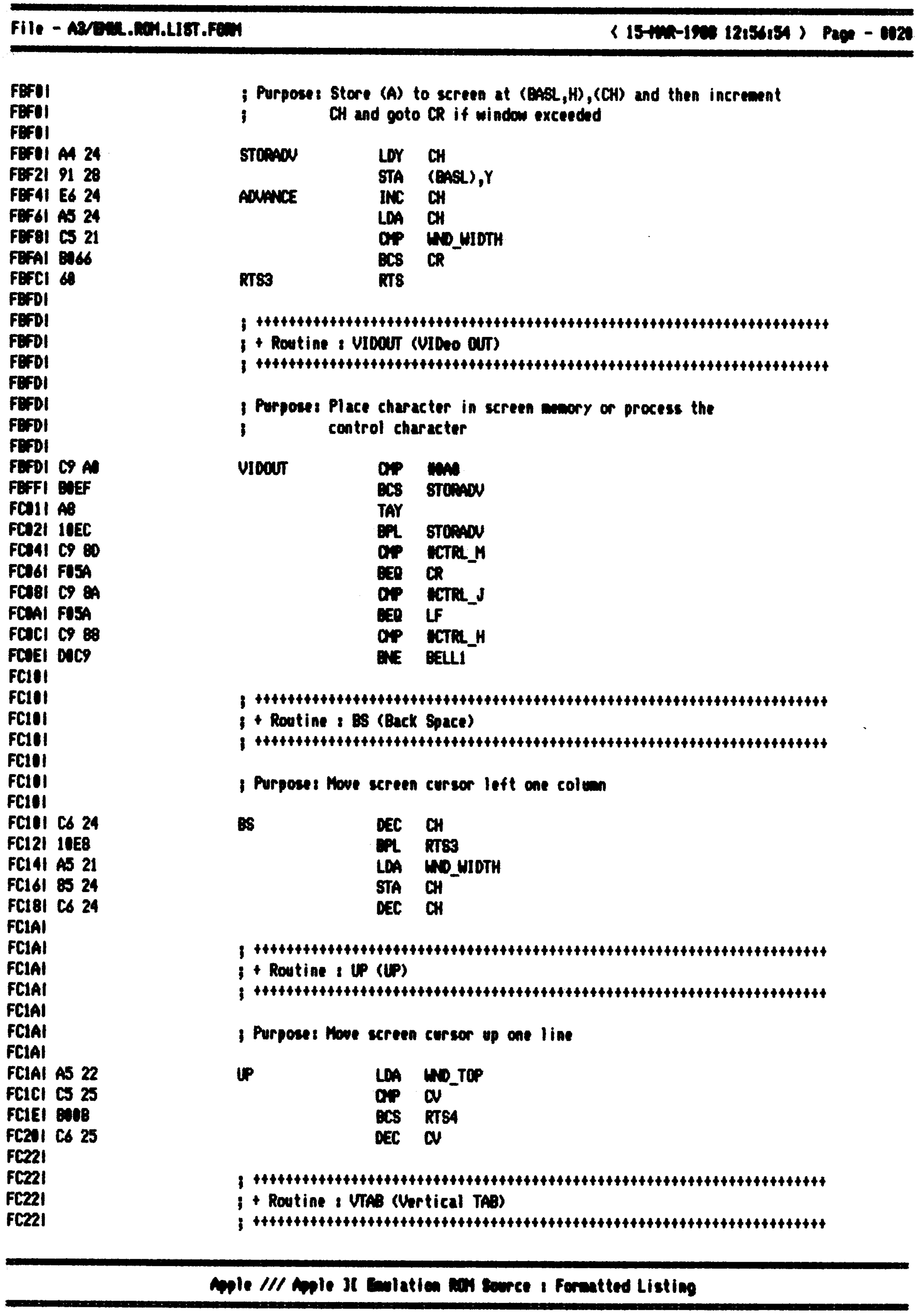


```
FB97|                      ; Purpose: Old Escape key handler
FB97|
FB97| 38                   ESCOLD        SEC
FB98| 4C 2CFC                             JMP   ESC1
FB9B|
FB9B|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB9B|                      ; + Routine : ESCNOW (ESCape NOW)
FB9B|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FB9B|
FB9B|                      ; Purpose: Handle Escape key press
FB9B|
FB9B| A8                   ESCNOW        TAY
FB9C| B9 48FA                             LDA   XLTBL-ASCII_I,Y
FB9F| 20 97FB                             JSR   ESCOLD
FBA2| 20 0CFD                             JSR   RDKEY
FBA5|
FBA5| C9 CE                ESCNEW        CMP   #ASCII_N
FBA7| B0EE                                BCS   ESCOLD
FBA9| C9 C9                               CMP   #ASCII_I
FBAB| 90EA                                BCC   ESCOLD
FBAD| C9 CC                               CMP   #ASCII_L
FBAF| F0E6                                BEQ   ESCOLD
FBB1| D0E8                                BNE   ESCNOW
FBB3|
FBB3|                      ; NOTE: Address $FBB3 contains an Identification Byte ($EA)
FBB3|
FBB3| EA                   F8VERSION     NOP            ; filler
FBB4| EA                                 NOP
FBB5| EA                                 NOP
FBB6| EA                                 NOP
FBB7| EA                                 NOP
FBB8| EA                                 NOP
FBB9| EA                                 NOP
FBBA| EA                                 NOP
FBBB| EA                                 NOP
FBBC| EA                                 NOP
FBBD| EA                                 NOP
FBBE| EA                                 NOP
FBBF| EA                                 NOP
FBC0| EA                                 NOP
FBC1|
FBC1|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBC1|                      ; + Routine : BASCALC (BASe address CALCulator)
FBC1|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBC1|
FBC1|                      ; Purpose: Set BASL,H to memory address for left character of line
FBC1|                      ;          in (A)
FBC1|
FBC1| 48                   BASCALC       PHA
FBC2| 4A                                 LSR   A
FBC3| 29 03                              AND   #3
FBC5| 09 04                              ORA   #4
FBC7| 85 29                              STA   BASH
FBC9| 68                                 PLA
FBCA| 29 18                              AND   #18
```

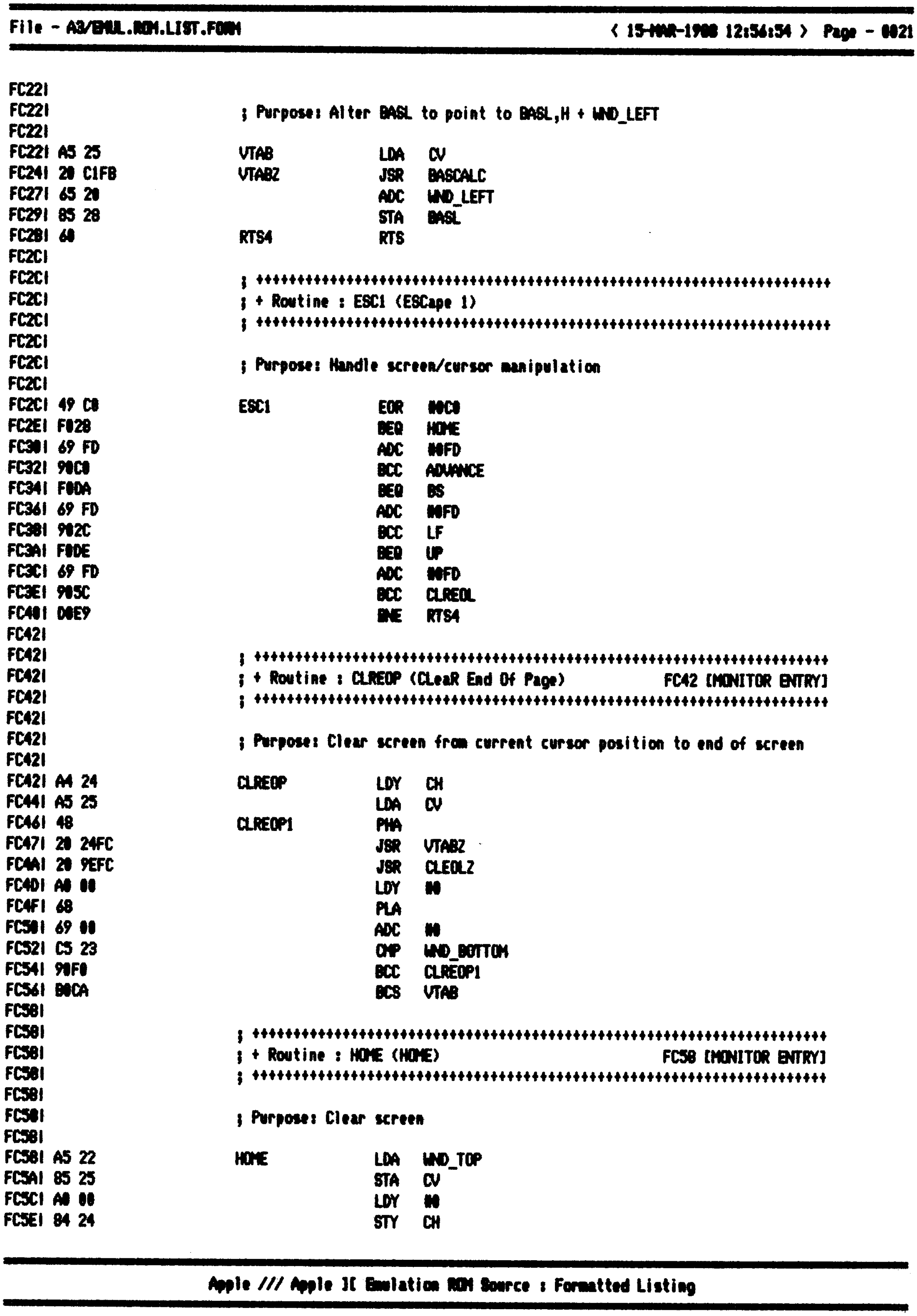


```
FBCC| 9002                              BCC   BASCLC2
FBCE| 69 7F                             ADC   #7F
FBD0| 85 28              BASCLC2        STA   BASL
FBD2| 0A                                ASL   A
FBD3| 0A                                ASL   A
FBD4| 05 28                             ORA   BASL
FBD6| 85 28                             STA   BASL
FBD8| 60                                RTS
FBD9|
FBD9|                    ; NOTE    : Screen output speed can dramatically be improved by
FBD9|                    ;           using the following routine in place of BASCALC:
FBD9|                    ;
FBD9|                    ; WARNING : This routine alters both the A and Y registers and
FBD9|                    ;           occupies more memory than BASCALC.
FBD9|                    ;
FBD9|                    ; FAST_BASCALC     ASL    A                    ; setup word index
FBD9|                    ;                  TAY
FBD9|                    ;                  LDA    (SCRN_ROW_TABLE),Y   ; get MSB of row address
FBD9|                    ;                  STA    BASH                 ;   from row table
FBD9|                    ;                  INY
FBD9|                    ;                  LDA    (SCRN_ROW_TABLE),Y   ; get LSB of row address
FBD9|                    ;                  STA    BASL                 ;   from row table
FBD9|                    ;                  RTS                         ; return to caller
FBD9|                    ;
FBD9|                    ; Screen row addresses for screen page 1 (Addresses $400-$7FF)
FBD9|                    ;
FBD9|                    ; SCRN_ROW_TABLE  .WORD  400,480,500,580      ; rows  0 -  3
FBD9|                    ;                 .WORD  600,680,700,780      ; rows  4 -  7
FBD9|                    ;                 .WORD  428,4A8,528,5A8      ; rows  8 - 11
FBD9|                    ;                 .WORD  628,6A8,728,7A8      ; rows 12 - 15
FBD9|                    ;                 .WORD  450,4D0,550,5D0      ; rows 16 - 19
FBD9|                    ;                 .WORD  650,6D0,750,7D0      ; rows 20 - 23
FBD9|
FBD9|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBD9|                    ; + Routine : BELL1 (BELL 1)                           FBD9 [MONITOR ENTRY]
FBD9|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBD9|
FBD9|                    ; Purpose: Toggle the built-in speaker to create a bell sound
FBD9|
FBD9| C9 87              BELL1          CMP   #CTRL_G
FBDB| D012                              BNE   RTS2B
FBDD| A9 40                             LDA   #40
FBDF| 20 A8FC                           JSR   WAIT
FBE2| A0 C0                             LDY   #C0
FBE4| A9 0C              BELL2          LDA   #0C
FBE6| 20 A8FC                           JSR   WAIT
FBE9| AD 30C0                           LDA   SPKR
FBEC| 88                                DEY
FBED| D0F5                              BNE   BELL2
FBEF| 60                 RTS2B          RTS
FBF0|
FBF0|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBF0|                    ; + Routine : STORADV (STORe and ADVance)
FBF0|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBF0|
```

"_23.PICT" 169 KB 2001-08-13 dpi: 300h x 300v pix: 2249h x 3166v

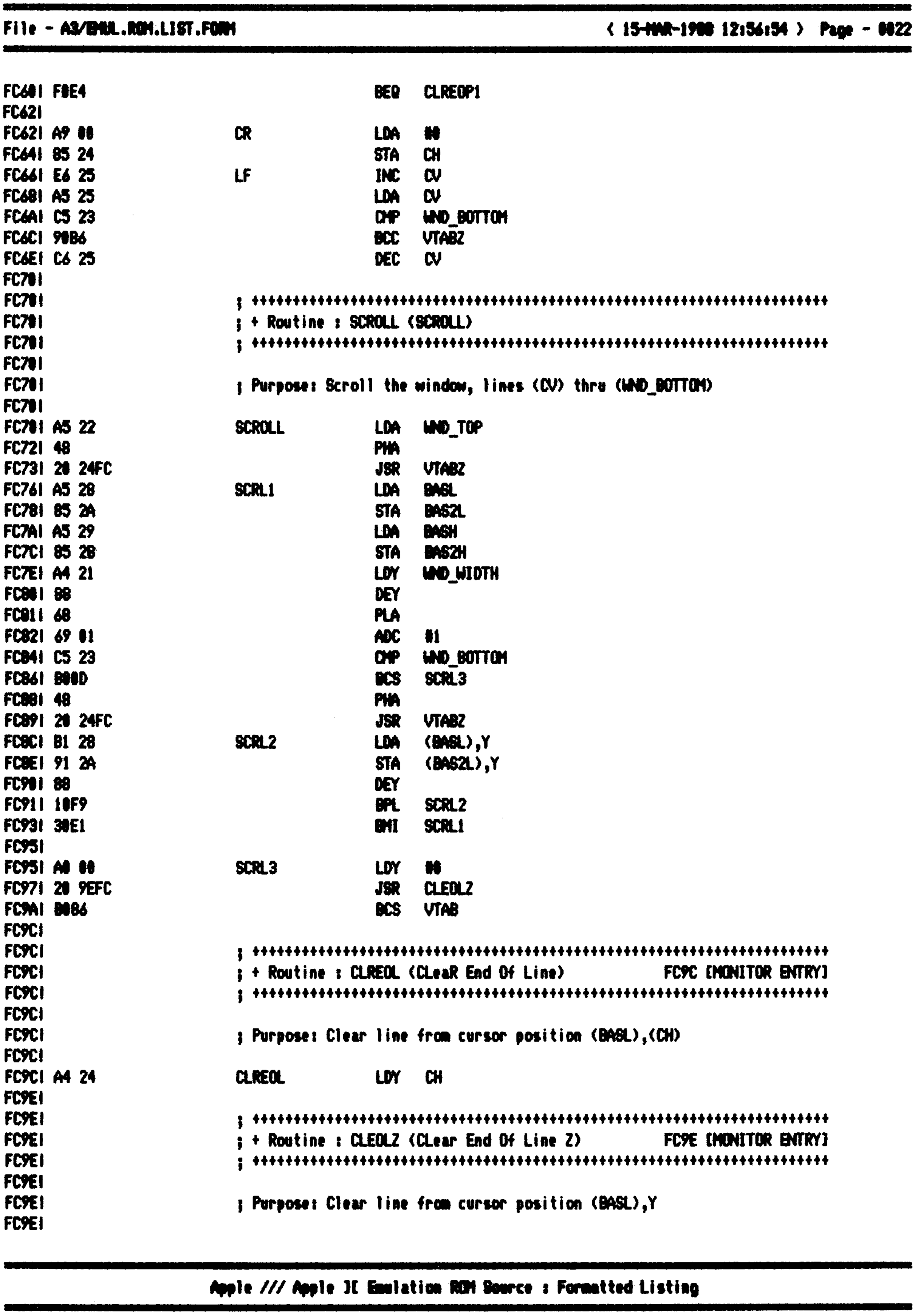


```
File - A3/EMUL.ROM.LIST.FORM                                  ( 15-MAR-1988 12:56:54 )  Page - 0020

FBF0|                    ; Purpose: Store (A) to screen at (BASL,H),(CH) and then increment
FBF0|                    ;          CH and goto CR if window exceeded
FBF0|
FBF0| A4 24              STORADV    LDY   CH
FBF2| 91 28                         STA   (BASL),Y
FBF4| E6 24              ADVANCE    INC   CH
FBF6| A5 24                         LDA   CH
FBF8| C5 21                         CMP   WND_WIDTH
FBFA| B066                          BCS   CR
FBFC| 60                 RTS3       RTS
FBFD|
FBFD|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBFD|                    ; + Routine : VIDOUT (VIDeo OUT)
FBFD|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FBFD|
FBFD|                    ; Purpose: Place character in screen memory or process the
FBFD|                    ;          control character
FBFD|
FBFD| C9 A0              VIDOUT     CMP   #$A0
FBFF| B0EF                          BCS   STORADV
FC01| A8                            TAY
FC02| 10EC                          BPL   STORADV
FC04| C9 8D                         CMP   #CTRL_M
FC06| F05A                          BEQ   CR
FC08| C9 8A                         CMP   #CTRL_J
FC0A| F05A                          BEQ   LF
FC0C| C9 88                         CMP   #CTRL_H
FC0E| D0C9                          BNE   BELL1
FC10|
FC10|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC10|                    ; + Routine : BS (Back Space)
FC10|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC10|
FC10|                    ; Purpose: Move screen cursor left one column
FC10|
FC10| C6 24              BS         DEC   CH
FC12| 10E8                          BPL   RTS3
FC14| A5 21                         LDA   WND_WIDTH
FC16| 85 24                         STA   CH
FC18| C6 24                         DEC   CH
FC1A|
FC1A|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC1A|                    ; + Routine : UP (UP)
FC1A|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC1A|
FC1A|                    ; Purpose: Move screen cursor up one line
FC1A|
FC1A| A5 22              UP         LDA   WND_TOP
FC1C| C5 25                         CMP   CV
FC1E| B00B                          BCS   RTS4
FC20| C6 25                         DEC   CV
FC22|
FC22|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC22|                    ; + Routine : VTAB (Vertical TAB)
FC22|                    ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                    Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```



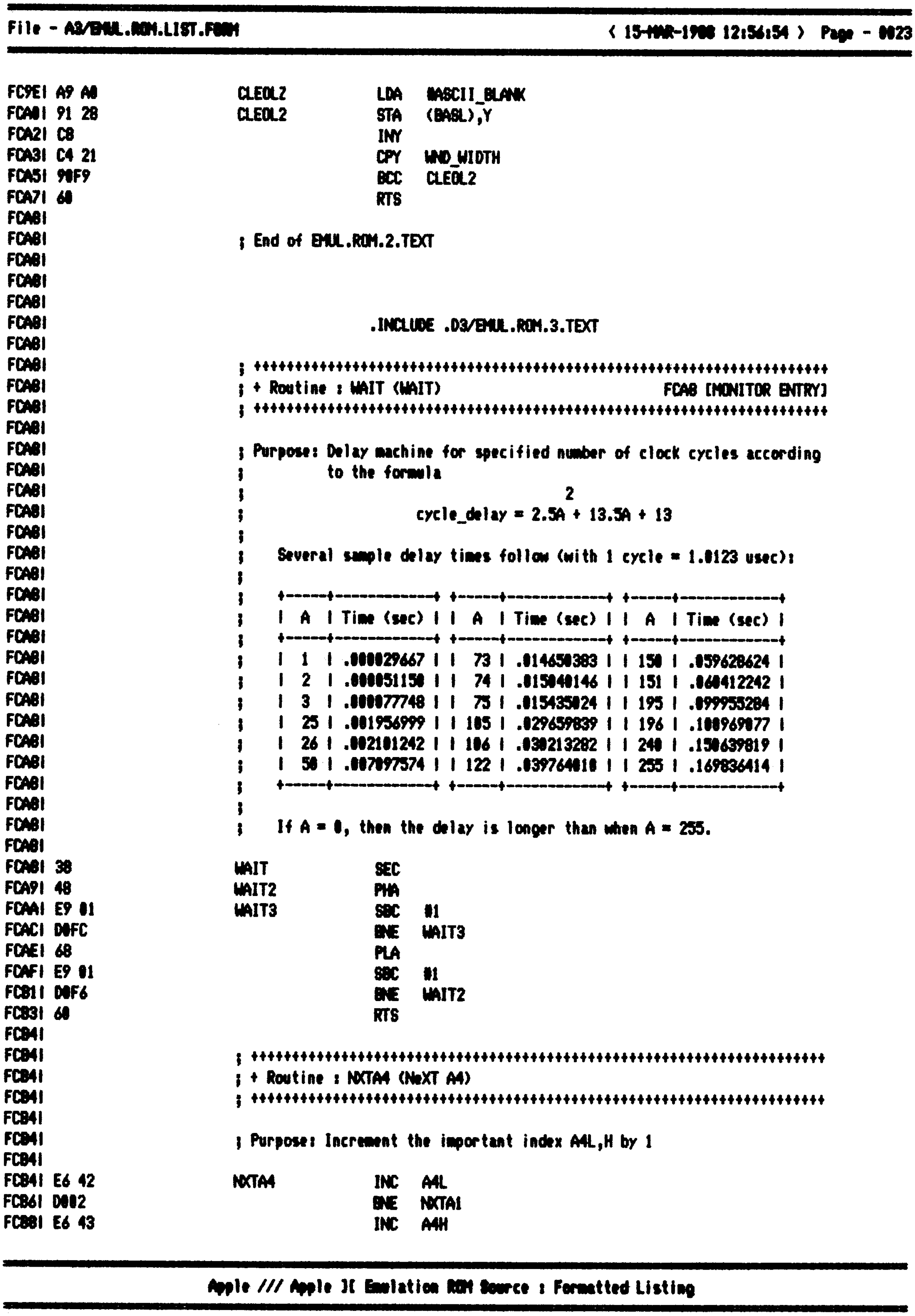
```
File - A3/EMUL.ROM.LIST.FORM                                    < 15-MAR-1980 12:54:54 >  Page - 0021

FC22|
FC22|                          ; Purpose: Alter BASL to point to BASL,H + WND_LEFT
FC22|
FC22| A5 25                    VTAB            LDA   CV
FC24| 20 C1FB                  VTABZ           JSR   BASCALC
FC27| 65 20                                    ADC   WND_LEFT
FC29| 85 28                                    STA   BASL
FC2B| 60                       RTS4            RTS
FC2C|
FC2C|                          ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC2C|                          ; + Routine : ESC1 (ESCape 1)
FC2C|                          ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC2C|
FC2C|                          ; Purpose: Handle screen/cursor manipulation
FC2C|
FC2C| 49 C0                    ESC1            EOR   #$C0
FC2E| F028                                     BEQ   HOME
FC30| 69 FD                                    ADC   #$FD
FC32| 90C0                                     BCC   ADVANCE
FC34| F0DA                                     BEQ   BS
FC36| 69 FD                                    ADC   #$FD
FC38| 902C                                     BCC   LF
FC3A| F0DE                                     BEQ   UP
FC3C| 69 FD                                    ADC   #$FD
FC3E| 905C                                     BCC   CLREOL
FC40| D0E9                                     BNE   RTS4
FC42|
FC42|                          ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC42|                          ; + Routine : CLREOP (CLeaR End Of Page)          FC42 [MONITOR ENTRY]
FC42|                          ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC42|
FC42|                          ; Purpose: Clear screen from current cursor position to end of screen
FC42|
FC42| A4 24                    CLREOP          LDY   CH
FC44| A5 25                                    LDA   CV
FC46| 48                       CLREOP1         PHA
FC47| 20 24FC                                  JSR   VTABZ
FC4A| 20 9EFC                                  JSR   CLEOLZ
FC4D| A0 00                                    LDY   #0
FC4F| 68                                       PLA
FC50| 69 00                                    ADC   #0
FC52| C5 23                                    CMP   WND_BOTTOM
FC54| 90F0                                     BCC   CLREOP1
FC56| B0CA                                     BCS   VTAB
FC58|
FC58|                          ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC58|                          ; + Routine : HOME (HOME)                         FC58 [MONITOR ENTRY]
FC58|                          ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC58|
FC58|                          ; Purpose: Clear screen
FC58|
FC58| A5 22                    HOME            LDA   WND_TOP
FC5A| 85 25                                    STA   CV
FC5C| A0 00                                    LDY   #0
FC5E| 84 24                                    STY   CH

                      Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

"_25.PICT" 153 KB 2001-08-13 dpi: 300h x 300v pix: 2225h x 3184v



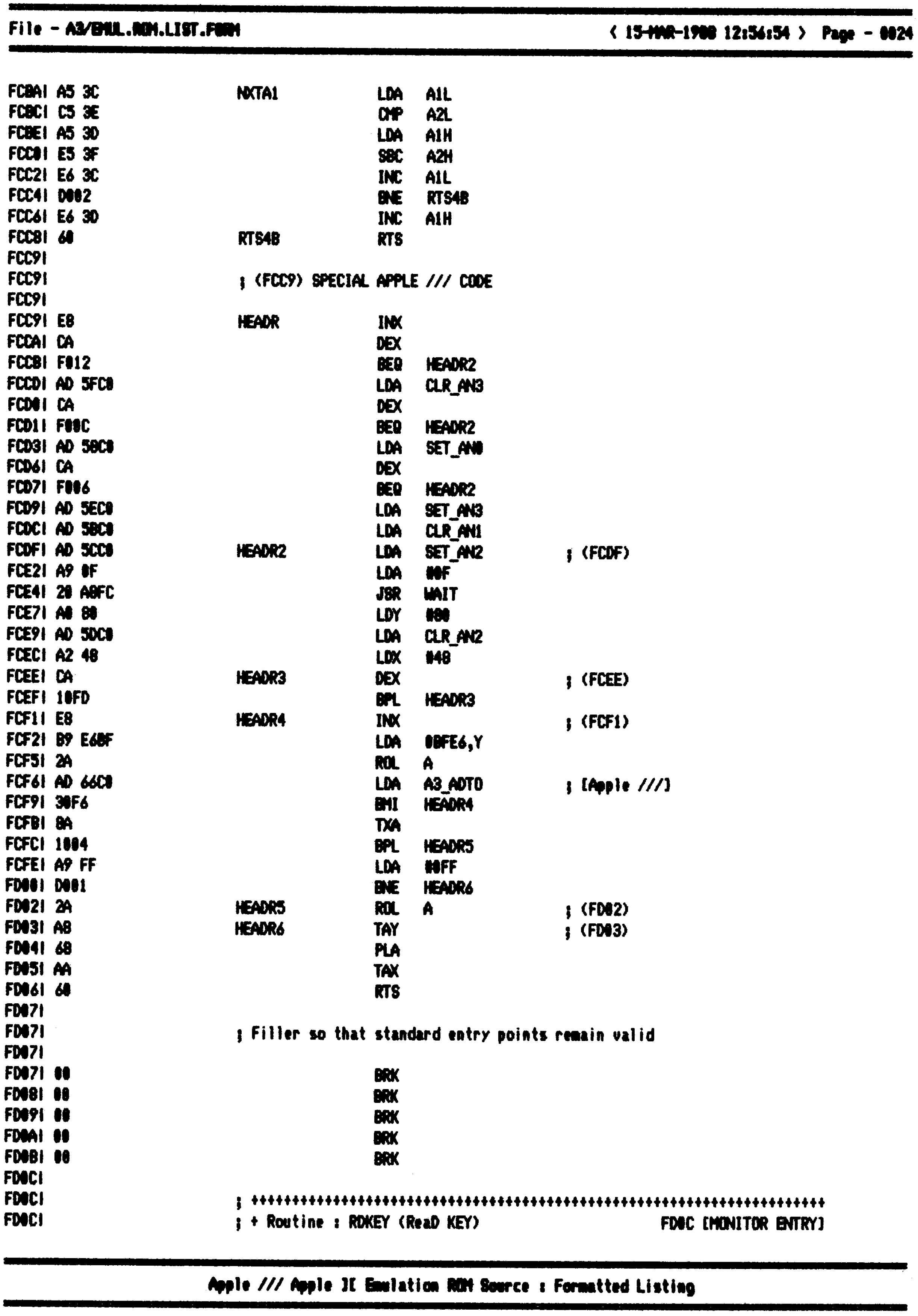
```
FC60| F0E4                          BEQ   CLREOP1
FC62|
FC62| A9 00          CR             LDA   #0
FC64| 85 24                         STA   CH
FC66| E6 25          LF             INC   CV
FC68| A5 25                         LDA   CV
FC6A| C5 23                         CMP   WND_BOTTOM
FC6C| 90B6                          BCC   VTABZ
FC6E| C6 25                         DEC   CV
FC70|
FC70|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC70|                ; + Routine : SCROLL (SCROLL)
FC70|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC70|
FC70|                ; Purpose: Scroll the window, lines (CV) thru (WND_BOTTOM)
FC70|
FC70| A5 22          SCROLL         LDA   WND_TOP
FC72| 48                            PHA
FC73| 20 24FC                       JSR   VTABZ
FC76| A5 28          SCRL1          LDA   BASL
FC78| 85 2A                         STA   BAS2L
FC7A| A5 29                         LDA   BASH
FC7C| 85 2B                         STA   BAS2H
FC7E| A4 21                         LDY   WND_WIDTH
FC80| 88                            DEY
FC81| 68                            PLA
FC82| 69 01                         ADC   #1
FC84| C5 23                         CMP   WND_BOTTOM
FC86| B00D                          BCS   SCRL3
FC88| 48                            PHA
FC89| 20 24FC                       JSR   VTABZ
FC8C| B1 28          SCRL2          LDA   (BASL),Y
FC8E| 91 2A                         STA   (BAS2L),Y
FC90| 88                            DEY
FC91| 10F9                          BPL   SCRL2
FC93| 30E1                          BMI   SCRL1
FC95|
FC95| A0 00          SCRL3          LDY   #0
FC97| 20 9EFC                       JSR   CLEOLZ
FC9A| B0B6                          BCS   VTAB
FC9C|
FC9C|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC9C|                ; + Routine : CLREOL (CLeaR End Of Line)          FC9C [MONITOR ENTRY]
FC9C|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC9C|
FC9C|                ; Purpose: Clear line from cursor position (BASL),(CH)
FC9C|
FC9C| A4 24          CLREOL         LDY   CH
FC9E|
FC9E|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC9E|                ; + Routine : CLEOLZ (CLear End Of Line Z)        FC9E [MONITOR ENTRY]
FC9E|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FC9E|
FC9E|                ; Purpose: Clear line from cursor position (BASL),Y
FC9E|
```

"_26.PICT" 153 KB 2001-08-13 dpi: 300h x 300v pix: 2249h x 3178v



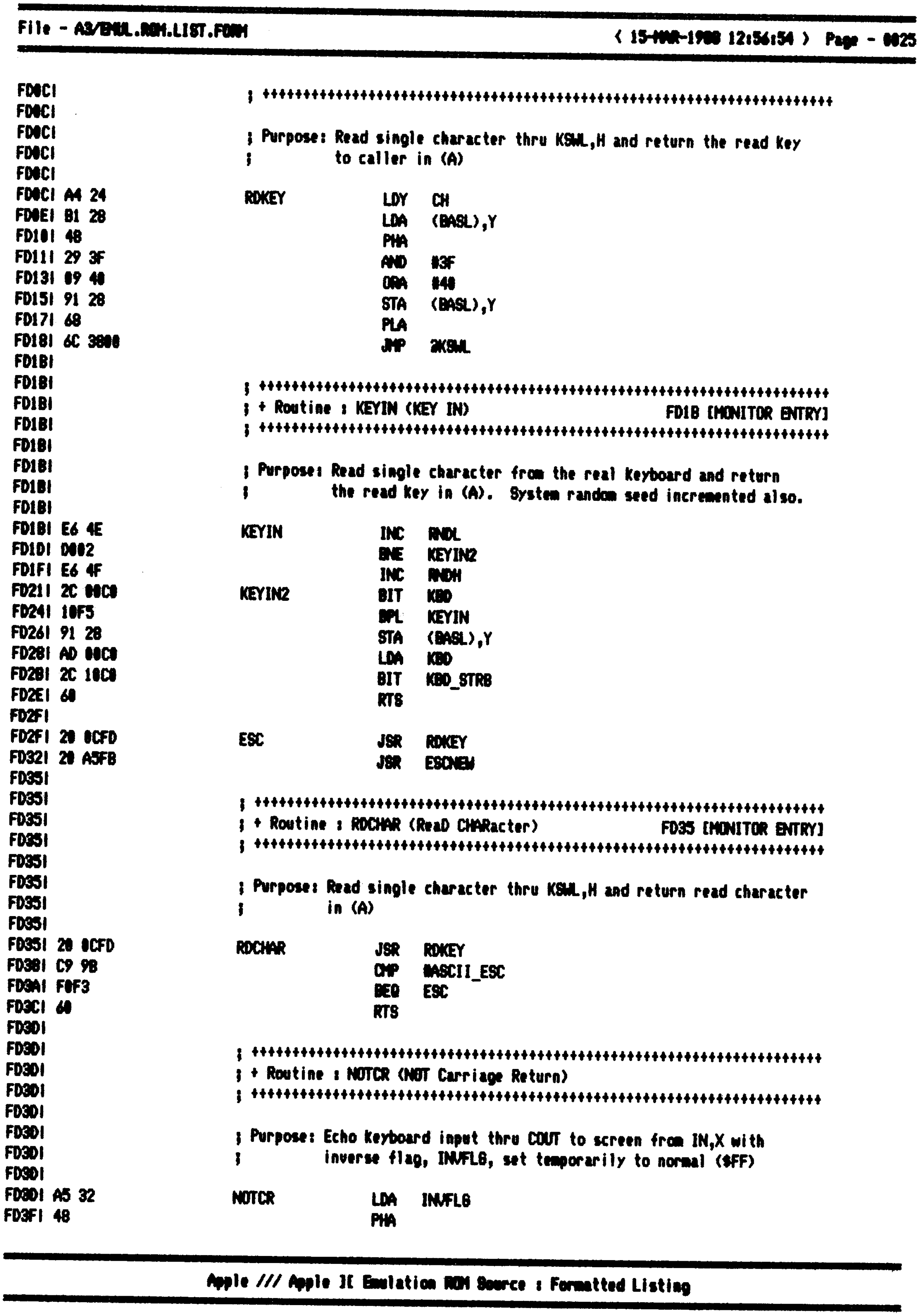
```
File - A3/EMUL.ROM.LIST.FORM                                    < 15-MAR-1980 12:56:54 >  Page - 0023

FC9E| A9 A0                CLEOLZ       LDA   #ASCII_BLANK
FCA0| 91 28                CLEOL2       STA   (BASL),Y
FCA2| C8                                INY
FCA3| C4 21                             CPY   WND_WIDTH
FCA5| 90F9                              BCC   CLEOL2
FCA7| 60                                RTS
FCA8|
FCA8|                      ; End of EMUL.ROM.2.TEXT
FCA8|
FCA8|
FCA8|
FCA8|                                  .INCLUDE .D3/EMUL.ROM.3.TEXT
FCA8|
FCA8|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FCA8|                      ; + Routine : WAIT (WAIT)                           FCA8 [MONITOR ENTRY]
FCA8|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FCA8|
FCA8|                      ; Purpose: Delay machine for specified number of clock cycles according
FCA8|                      ;          to the formula
FCA8|                      ;                                         2
FCA8|                      ;                     cycle_delay = 2.5A + 13.5A + 13
FCA8|                      ;
FCA8|                      ;   Several sample delay times follow (with 1 cycle = 1.0123 usec):
FCA8|                      ;
FCA8|                      ;   +-----+------------+ +-----+------------+ +-----+------------+
FCA8|                      ;   |  A  | Time (sec) | |  A  | Time (sec) | |  A  | Time (sec) |
FCA8|                      ;   +-----+------------+ +-----+------------+ +-----+------------+
FCA8|                      ;   |  1  | .000029667 | |  73 | .014650383 | | 150 | .059628624 |
FCA8|                      ;   |  2  | .000051150 | |  74 | .015040146 | | 151 | .060412242 |
FCA8|                      ;   |  3  | .000077748 | |  75 | .015435024 | | 195 | .099955204 |
FCA8|                      ;   |  25 | .001956999 | | 105 | .029659839 | | 196 | .100969077 |
FCA8|                      ;   |  26 | .002101242 | | 106 | .030213282 | | 240 | .150639819 |
FCA8|                      ;   |  50 | .007097574 | | 122 | .039764010 | | 255 | .169836414 |
FCA8|                      ;   +-----+------------+ +-----+------------+ +-----+------------+
FCA8|                      ;
FCA8|                      ;   If A = 0, then the delay is longer than when A = 255.
FCA8|
FCA8| 38                   WAIT         SEC
FCA9| 48                   WAIT2        PHA
FCAA| E9 01                WAIT3        SBC   #1
FCAC| D0FC                              BNE   WAIT3
FCAE| 68                                PLA
FCAF| E9 01                             SBC   #1
FCB1| D0F6                              BNE   WAIT2
FCB3| 60                                RTS
FCB4|
FCB4|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FCB4|                      ; + Routine : NXTA4 (NeXT A4)
FCB4|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FCB4|
FCB4|                      ; Purpose: Increment the important index A4L,H by 1
FCB4|
FCB4| E6 42                NXTA4        INC   A4L
FCB6| D002                              BNE   NXTA1
FCB8| E6 43                             INC   A4H

                 Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

“_27.PICT” 171 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3178v



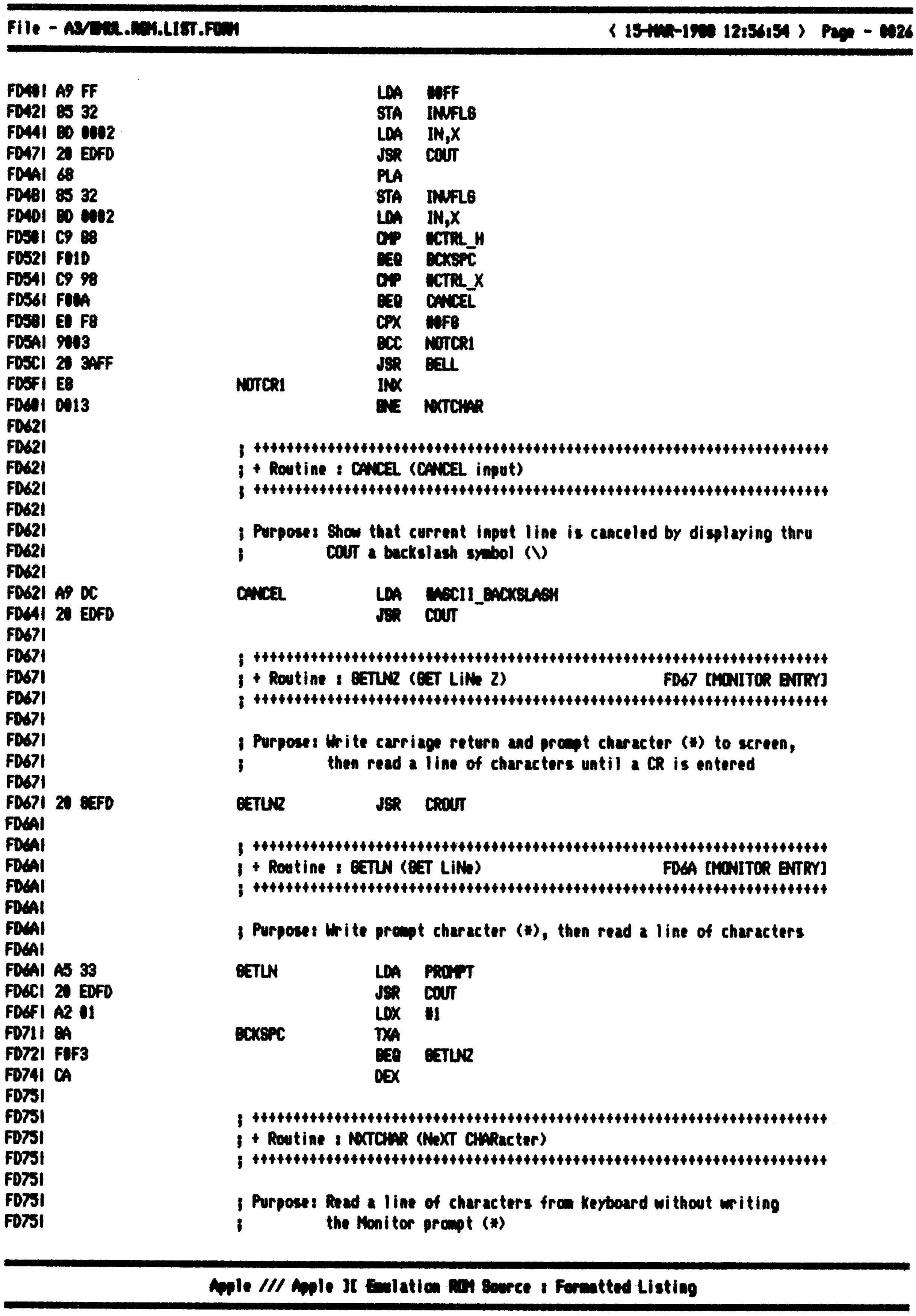
```
FCBA| A5 3C         NXTA1        LDA   A1L
FCBC| C5 3E                      CMP   A2L
FCBE| A5 3D                      LDA   A1H
FCC0| E5 3F                      SBC   A2H
FCC2| E6 3C                      INC   A1L
FCC4| D002                       BNE   RTS4B
FCC6| E6 3D                      INC   A1H
FCC8| 60            RTS4B        RTS
FCC9|
FCC9|               ; (FCC9) SPECIAL APPLE /// CODE
FCC9|
FCC9| E8            HEADR        INX
FCCA| CA                         DEX
FCCB| F012                       BEQ   HEADR2
FCCD| AD 5FC0                    LDA   CLR_AN3
FCD0| CA                         DEX
FCD1| F00C                       BEQ   HEADR2
FCD3| AD 58C0                    LDA   SET_AN0
FCD6| CA                         DEX
FCD7| F006                       BEQ   HEADR2
FCD9| AD 5EC0                    LDA   SET_AN3
FCDC| AD 5BC0                    LDA   CLR_AN1
FCDF| AD 5CC0       HEADR2       LDA   SET_AN2          ; (FCDF)
FCE2| A9 0F                      LDA   #0F
FCE4| 20 A8FC                    JSR   WAIT
FCE7| A0 80                      LDY   #80
FCE9| AD 5DC0                    LDA   CLR_AN2
FCEC| A2 48                      LDX   #48
FCEE| CA            HEADR3       DEX                    ; (FCEE)
FCEF| 10FD                       BPL   HEADR3
FCF1| E8            HEADR4       INX                    ; (FCF1)
FCF2| B9 E6BF                    LDA   0BFE6,Y
FCF5| 2A                         ROL   A
FCF6| AD 66C0                    LDA   A3_ADTO          ; [Apple ///]
FCF9| 30F6                       BMI   HEADR4
FCFB| 8A                         TXA
FCFC| 1004                       BPL   HEADR5
FCFE| A9 FF                      LDA   #0FF
FD00| D001                       BNE   HEADR6
FD02| 2A            HEADR5       ROL   A                ; (FD02)
FD03| A8            HEADR6       TAY                    ; (FD03)
FD04| 68                         PLA
FD05| AA                         TAX
FD06| 60                         RTS
FD07|
FD07|               ; Filler so that standard entry points remain valid
FD07|
FD07| 00                         BRK
FD08| 00                         BRK
FD09| 00                         BRK
FD0A| 00                         BRK
FD0B| 00                         BRK
FD0C|
FD0C|               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD0C|               ; + Routine : RDKEY (ReaD KEY)                    FD0C [MONITOR ENTRY]
```

“_28.PICT” 133 KB 2001-08-13 dpi: 300h x 300v pix: 2237h x 3172v



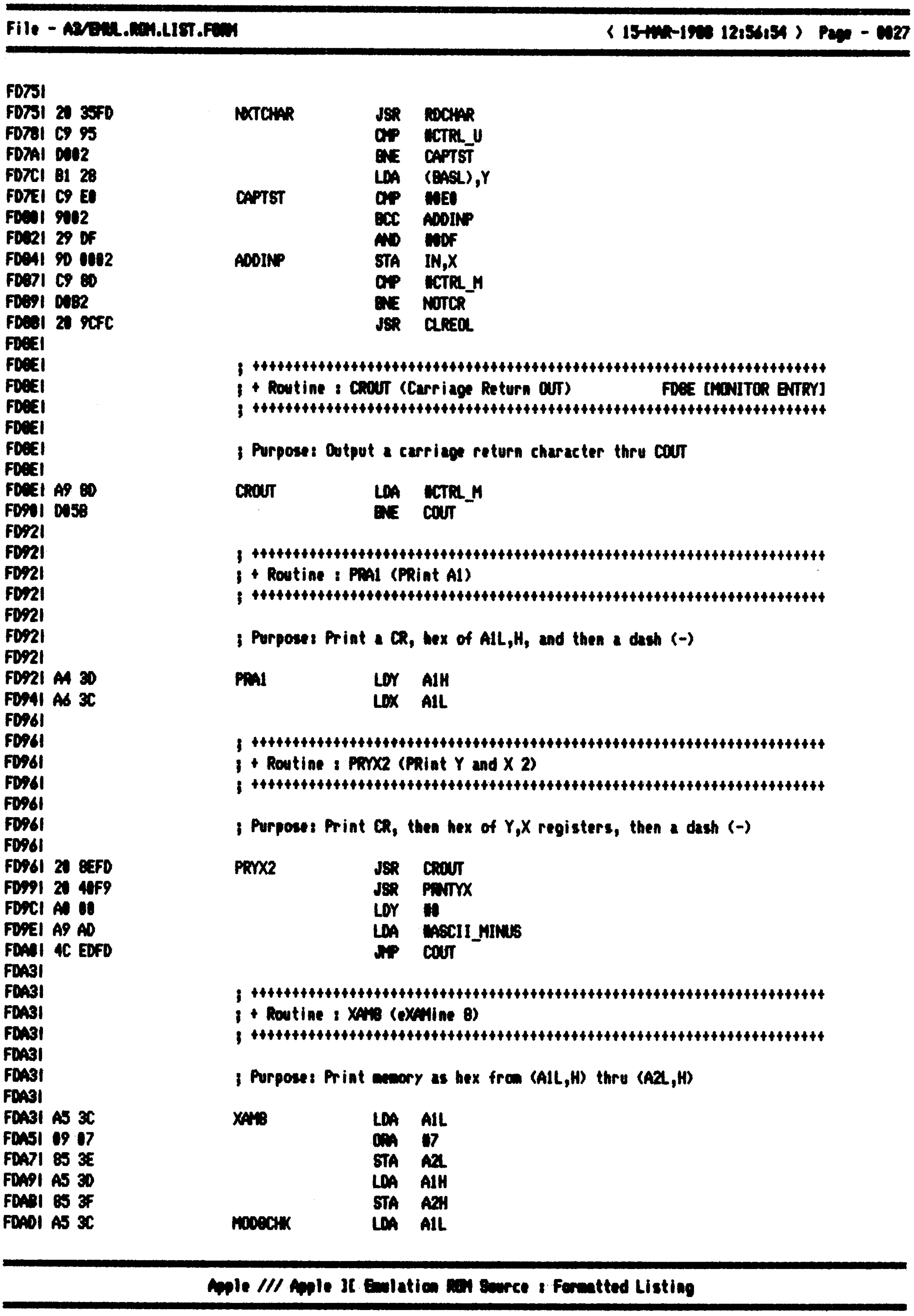
```
FD0C|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD0C|
FD0C|                      ; Purpose: Read single character thru KSWL,H and return the read key
FD0C|                      ;          to caller in (A)
FD0C|
FD0C| A4 24                RDKEY            LDY   CH
FD0E| B1 28                                 LDA   (BASL),Y
FD10| 48                                    PHA
FD11| 29 3F                                 AND   #3F
FD13| 09 40                                 ORA   #40
FD15| 91 28                                 STA   (BASL),Y
FD17| 68                                    PLA
FD18| 6C 3800                               JMP   @KSWL
FD1B|
FD1B|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD1B|                      ; + Routine : KEYIN (KEY IN)                          FD1B [MONITOR ENTRY]
FD1B|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD1B|
FD1B|                      ; Purpose: Read single character from the real keyboard and return
FD1B|                      ;          the read key in (A).  System random seed incremented also.
FD1B|
FD1B| E6 4E                KEYIN            INC   RNDL
FD1D| D002                                  BNE   KEYIN2
FD1F| E6 4F                                 INC   RNDH
FD21| 2C 00C0              KEYIN2           BIT   KBD
FD24| 10F5                                  BPL   KEYIN
FD26| 91 28                                 STA   (BASL),Y
FD28| AD 00C0                               LDA   KBD
FD2B| 2C 10C0                               BIT   KBD_STRB
FD2E| 60                                    RTS
FD2F|
FD2F| 20 0CFD              ESC              JSR   RDKEY
FD32| 20 A5FB                               JSR   ESCNEW
FD35|
FD35|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD35|                      ; + Routine : RDCHAR (ReaD CHARacter)                 FD35 [MONITOR ENTRY]
FD35|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD35|
FD35|                      ; Purpose: Read single character thru KSWL,H and return read character
FD35|                      ;          in (A)
FD35|
FD35| 20 0CFD              RDCHAR           JSR   RDKEY
FD38| C9 9B                                 CMP   #ASCII_ESC
FD3A| F0F3                                  BEQ   ESC
FD3C| 60                                    RTS
FD3D|
FD3D|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD3D|                      ; + Routine : NOTCR (NOT Carriage Return)
FD3D|                      ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD3D|
FD3D|                      ; Purpose: Echo keyboard input thru COUT to screen from IN,X with
FD3D|                      ;          inverse flag, INVFLG, set temporarily to normal ($FF)
FD3D|
FD3D| A5 32                NOTCR            LDA   INVFLG
FD3F| 48                                    PHA
```

"_29.PICT" 167 KB 2001-08-13 dpi: 300h x 300v pix: 2249h x 3184v



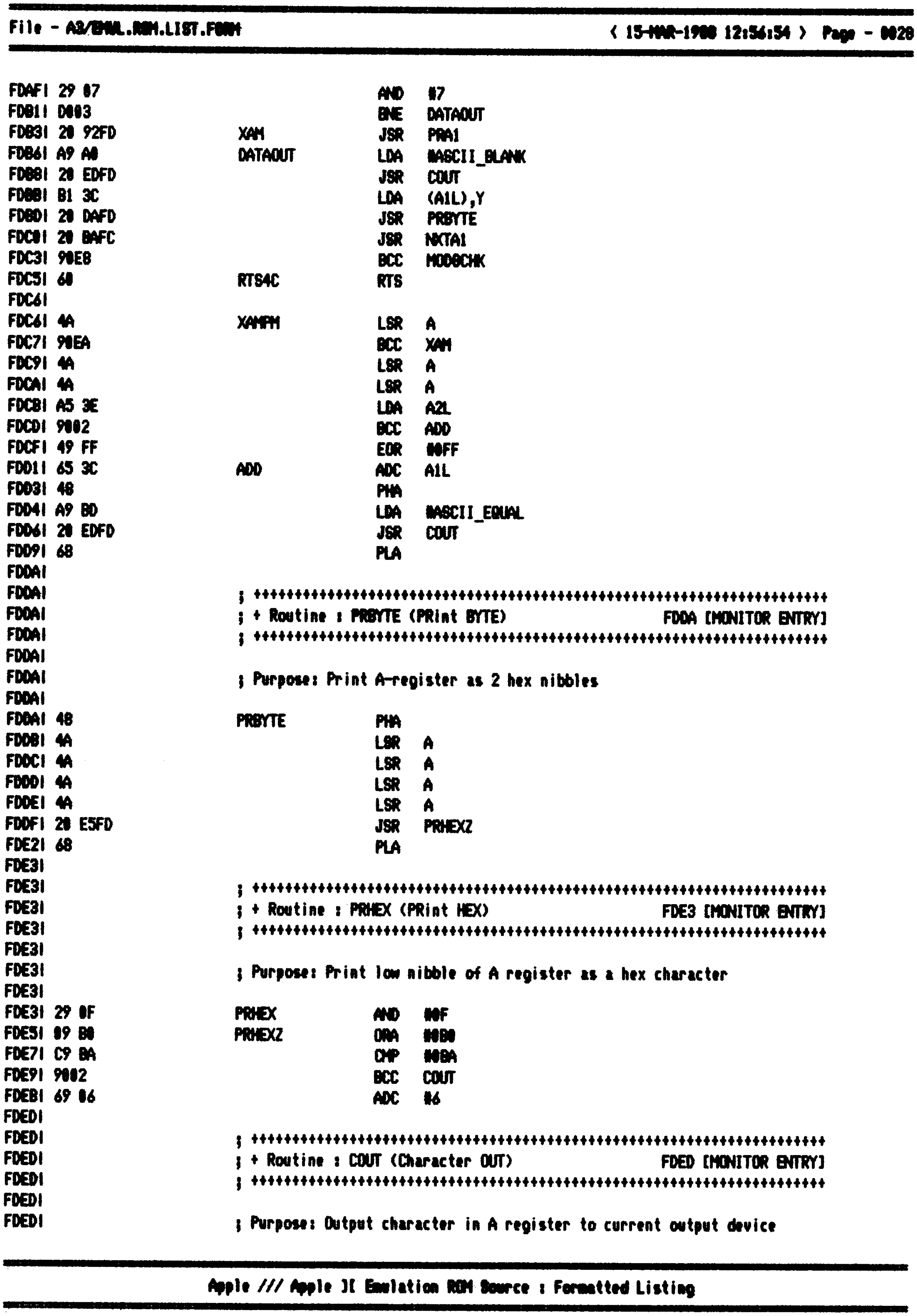
```
FD40| A9 FF                     LDA   #0FF
FD42| 85 32                     STA   INVFLG
FD44| BD 0002                   LDA   IN,X
FD47| 20 EDFD                   JSR   COUT
FD4A| 68                        PLA
FD4B| 85 32                     STA   INVFLG
FD4D| BD 0002                   LDA   IN,X
FD50| C9 88                     CMP   #CTRL_H
FD52| F01D                      BEQ   BCKSPC
FD54| C9 98                     CMP   #CTRL_X
FD56| F00A                      BEQ   CANCEL
FD58| E0 F8                     CPX   #0F8
FD5A| 9003                      BCC   NOTCR1
FD5C| 20 3AFF                   JSR   BELL
FD5F| E8          NOTCR1        INX
FD60| D013                      BNE   NXTCHAR
FD62|
FD62|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD62|             ; + Routine : CANCEL (CANCEL input)
FD62|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD62|
FD62|             ; Purpose: Show that current input line is canceled by displaying thru
FD62|             ;          COUT a backslash symbol (\)
FD62|
FD62| A9 DC       CANCEL        LDA   #ASCII_BACKSLASH
FD64| 20 EDFD                   JSR   COUT
FD67|
FD67|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD67|             ; + Routine : GETLNZ (GET LiNe Z)                FD67 [MONITOR ENTRY]
FD67|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD67|
FD67|             ; Purpose: Write carriage return and prompt character (*) to screen,
FD67|             ;          then read a line of characters until a CR is entered
FD67|
FD67| 20 8EFD     GETLNZ        JSR   CROUT
FD6A|
FD6A|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD6A|             ; + Routine : GETLN (GET LiNe)                   FD6A [MONITOR ENTRY]
FD6A|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD6A|
FD6A|             ; Purpose: Write prompt character (*), then read a line of characters
FD6A|
FD6A| A5 33       GETLN         LDA   PROMPT
FD6C| 20 EDFD                   JSR   COUT
FD6F| A2 01                     LDX   #1
FD71| 8A          BCKSPC        TXA
FD72| F0F3                      BEQ   GETLNZ
FD74| CA                        DEX
FD75|
FD75|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD75|             ; + Routine : NXTCHAR (NeXT CHARacter)
FD75|             ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD75|
FD75|             ; Purpose: Read a line of characters from keyboard without writing
FD75|             ;          the Monitor prompt (*)
```

"_30.PICT" 172 KB 2001-08-13 dpi: 300h x 300v pix: 2213h x 3166v



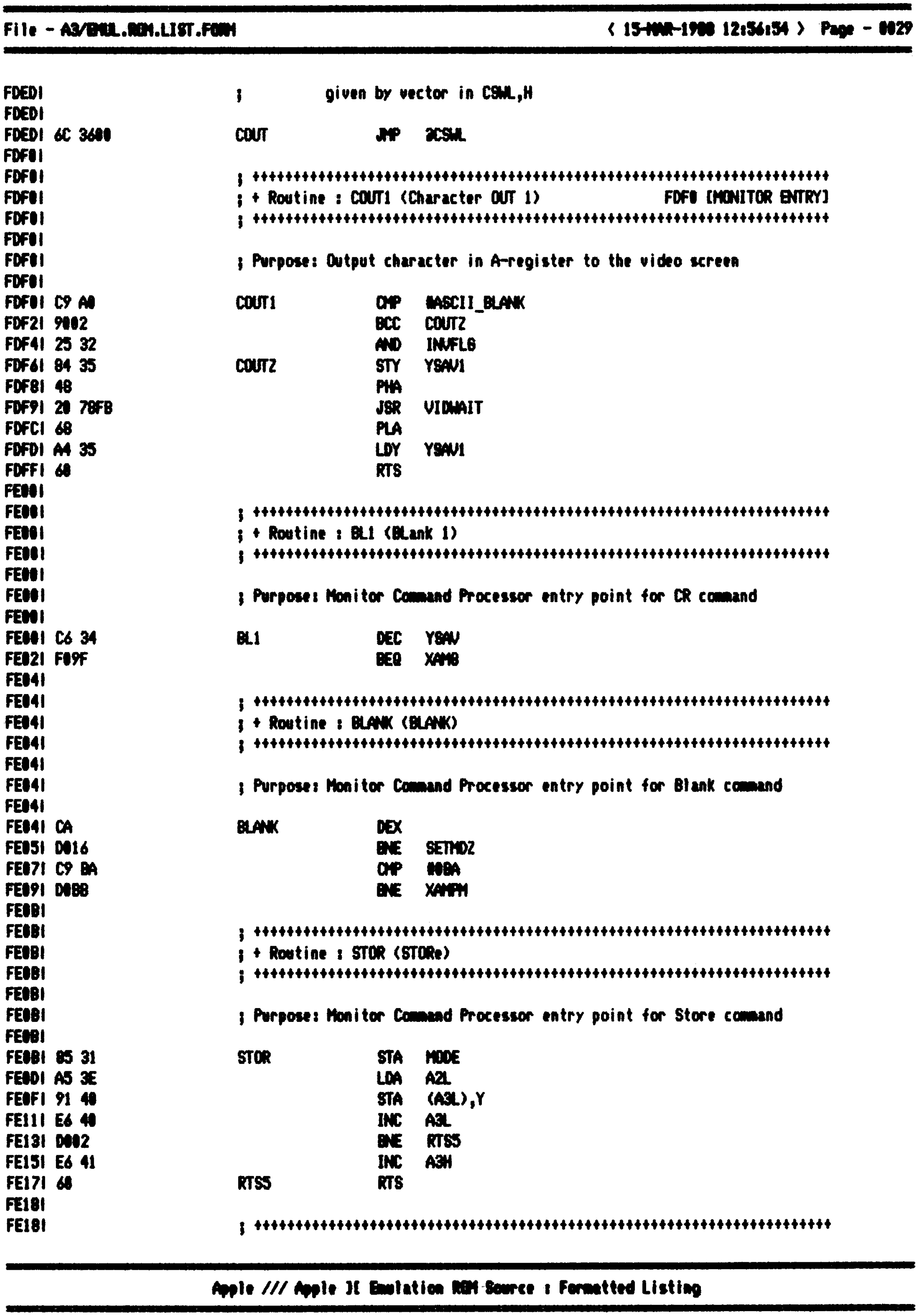
```
FD75|
FD75| 20 35FD          NXTCHAR        JSR   RDCHAR
FD78| C9 95                           CMP   #CTRL_U
FD7A| D002                            BNE   CAPTST
FD7C| B1 28                           LDA   (BASL),Y
FD7E| C9 E0            CAPTST         CMP   #0E0
FD80| 9002                            BCC   ADDINP
FD82| 29 DF                           AND   #0DF
FD84| 9D 0002          ADDINP         STA   IN,X
FD87| C9 8D                           CMP   #CTRL_M
FD89| D0B2                            BNE   NOTCR
FD8B| 20 9CFC                         JSR   CLREOL
FD8E|
FD8E|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD8E|                  ; + Routine : CROUT (Carriage Return OUT)          FD8E [MONITOR ENTRY]
FD8E|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD8E|
FD8E|                  ; Purpose: Output a carriage return character thru COUT
FD8E|
FD8E| A9 8D            CROUT          LDA   #CTRL_M
FD90| D05B                            BNE   COUT
FD92|
FD92|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD92|                  ; + Routine : PRA1 (PRint A1)
FD92|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD92|
FD92|                  ; Purpose: Print a CR, hex of A1L,H, and then a dash (-)
FD92|
FD92| A4 3D            PRA1           LDY   A1H
FD94| A6 3C                           LDX   A1L
FD96|
FD96|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD96|                  ; + Routine : PRYX2 (PRint Y and X 2)
FD96|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FD96|
FD96|                  ; Purpose: Print CR, then hex of Y,X registers, then a dash (-)
FD96|
FD96| 20 8EFD          PRYX2          JSR   CROUT
FD99| 20 40F9                         JSR   PRNTYX
FD9C| A0 00                           LDY   #0
FD9E| A9 AD                           LDA   #ASCII_MINUS
FDA0| 4C EDFD                         JMP   COUT
FDA3|
FDA3|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDA3|                  ; + Routine : XAM8 (eXAMine 8)
FDA3|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDA3|
FDA3|                  ; Purpose: Print memory as hex from (A1L,H) thru (A2L,H)
FDA3|
FDA3| A5 3C            XAM8           LDA   A1L
FDA5| 09 07                           ORA   #7
FDA7| 85 3E                           STA   A2L
FDA9| A5 3D                           LDA   A1H
FDAB| 85 3F                           STA   A2H
FDAD| A5 3C            MOD8CHK        LDA   A1L
```

“_31.PICT” 165 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3178v



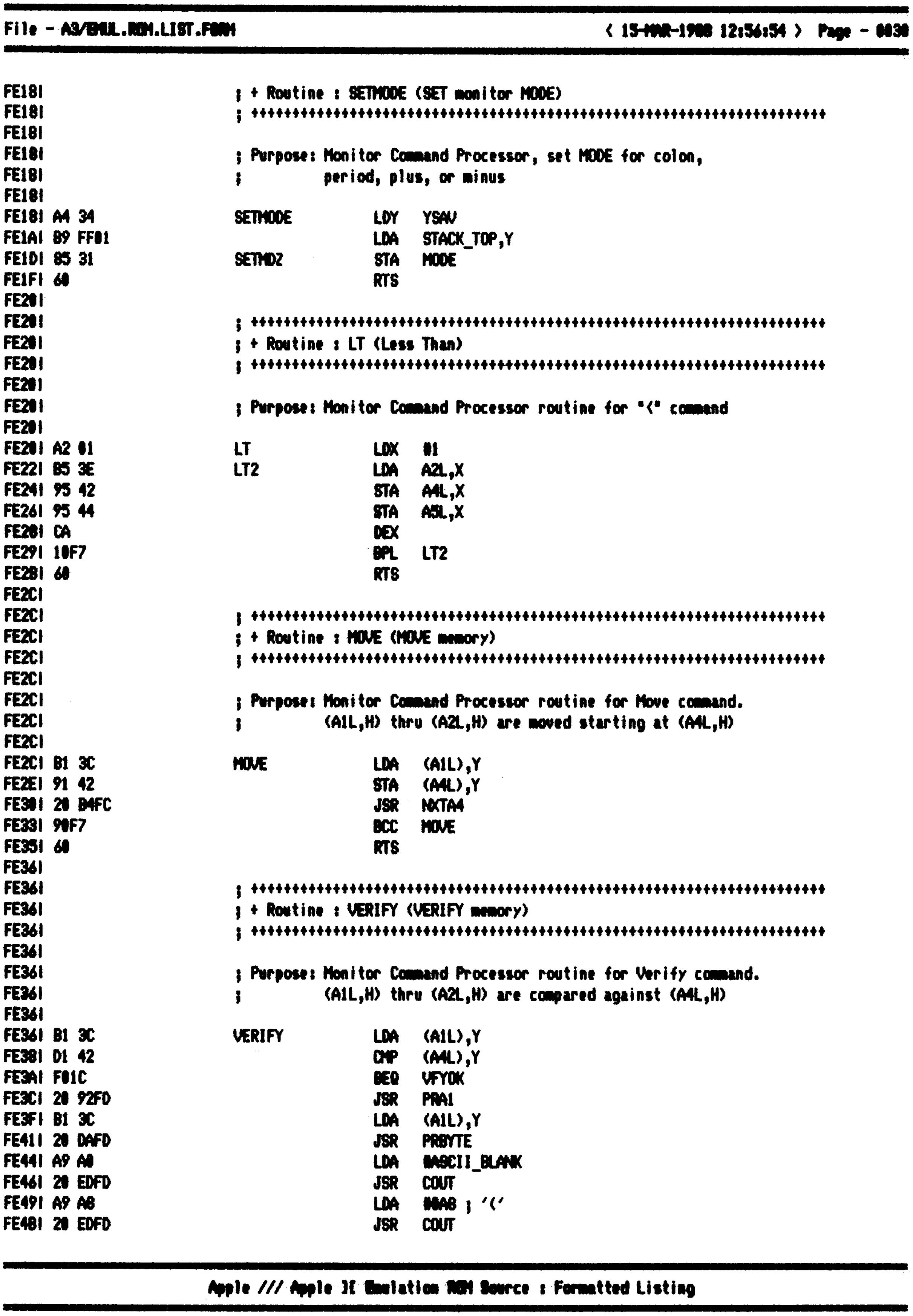
```
FDAF| 29 07                                 AND   #7
FDB1| D003                                  BNE   DATAOUT
FDB3| 20 92FD               XAM             JSR   PRA1
FDB6| A9 A0                 DATAOUT         LDA   #ASCII_BLANK
FDB8| 20 EDFD                               JSR   COUT
FDBB| B1 3C                                 LDA   (A1L),Y
FDBD| 20 DAFD                               JSR   PRBYTE
FDC0| 20 BAFC                               JSR   NXTA1
FDC3| 90E8                                  BCC   MOD8CHK
FDC5| 60                    RTS4C           RTS
FDC6|
FDC6| 4A                    XAMPM           LSR   A
FDC7| 90EA                                  BCC   XAM
FDC9| 4A                                    LSR   A
FDCA| 4A                                    LSR   A
FDCB| A5 3E                                 LDA   A2L
FDCD| 9002                                  BCC   ADD
FDCF| 49 FF                                 EOR   #0FF
FDD1| 65 3C                 ADD             ADC   A1L
FDD3| 48                                    PHA
FDD4| A9 BD                                 LDA   #ASCII_EQUAL
FDD6| 20 EDFD                               JSR   COUT
FDD9| 68                                    PLA
FDDA|
FDDA|                       ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDDA|                       ; + Routine : PRBYTE (PRint BYTE)                    FDDA [MONITOR ENTRY]
FDDA|                       ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDDA|
FDDA|                       ; Purpose: Print A-register as 2 hex nibbles
FDDA|
FDDA| 48                    PRBYTE          PHA
FDDB| 4A                                    LSR   A
FDDC| 4A                                    LSR   A
FDDD| 4A                                    LSR   A
FDDE| 4A                                    LSR   A
FDDF| 20 E5FD                               JSR   PRHEXZ
FDE2| 68                                    PLA
FDE3|
FDE3|                       ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDE3|                       ; + Routine : PRHEX (PRint HEX)                      FDE3 [MONITOR ENTRY]
FDE3|                       ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDE3|
FDE3|                       ; Purpose: Print low nibble of A register as a hex character
FDE3|
FDE3| 29 0F                 PRHEX           AND   #0F
FDE5| 09 B0                 PRHEXZ          ORA   #0B0
FDE7| C9 BA                                 CMP   #0BA
FDE9| 9002                                  BCC   COUT
FDEB| 69 06                                 ADC   #6
FDED|
FDED|                       ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDED|                       ; + Routine : COUT (Character OUT)                   FDED [MONITOR ENTRY]
FDED|                       ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDED|
FDED|                       ; Purpose: Output character in A register to current output device
```

"_32.PICT" 154 KB 2001-08-13 dpi: 300h x 300v pix: 2243h x 3166v

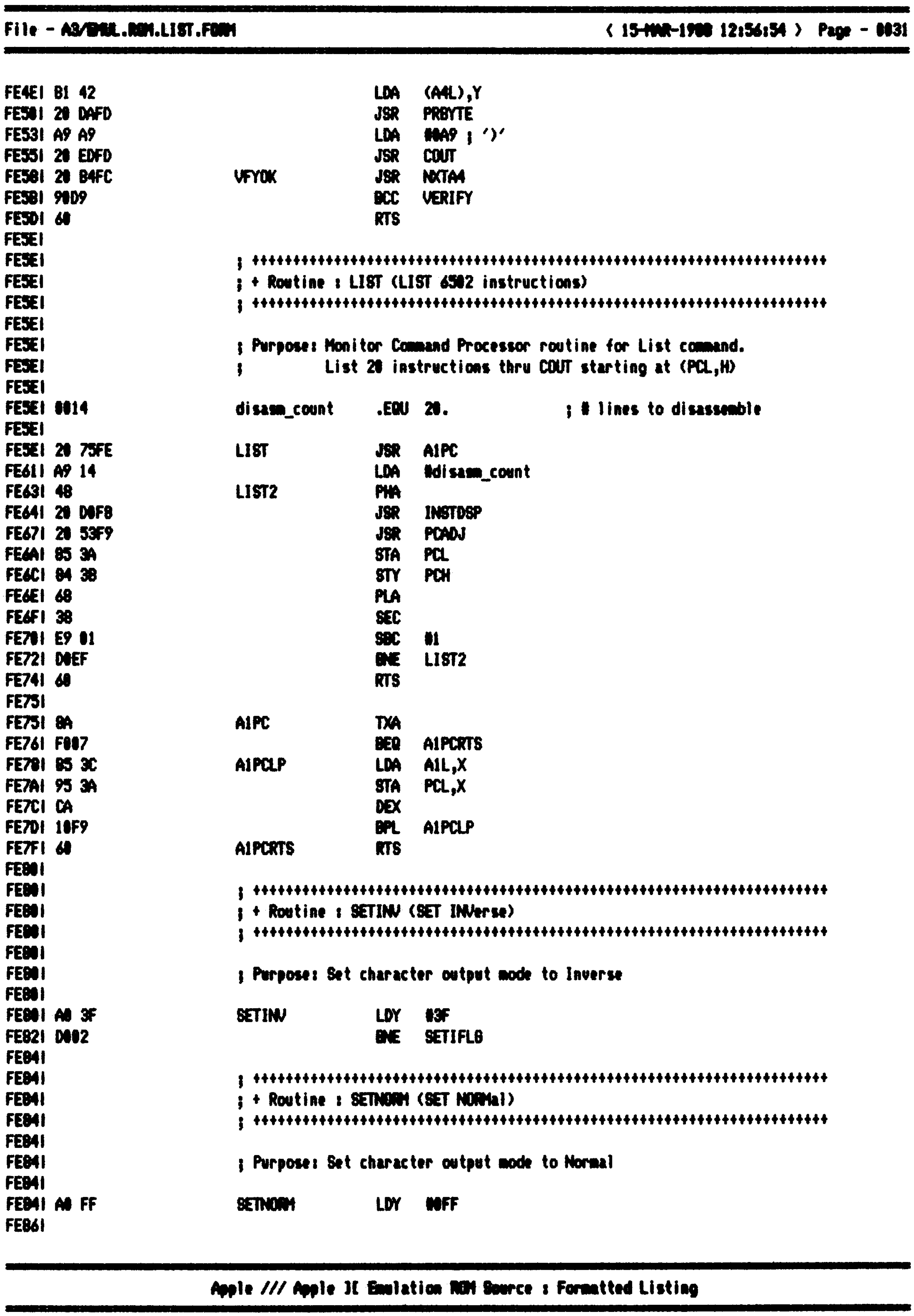


```
File - A3/EMUL.ROM.LIST.FORM                                   < 15-MAR-1988 12:56:54 >  Page - 0029

FDED|                   ;         given by vector in CSWL,H
FDED|
FDED| 6C 3600           COUT            JMP   @CSWL
FDF0|
FDF0|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDF0|                   ; + Routine : COUT1 (Character OUT 1)             FDF0 [MONITOR ENTRY]
FDF0|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FDF0|
FDF0|                   ; Purpose: Output character in A-register to the video screen
FDF0|
FDF0| C9 A0             COUT1           CMP   #ASCII_BLANK
FDF2| 9002                              BCC   COUTZ
FDF4| 25 32                             AND   INVFLG
FDF6| 84 35             COUTZ           STY   YSAV1
FDF8| 48                                PHA
FDF9| 20 78FB                           JSR   VIDWAIT
FDFC| 68                                PLA
FDFD| A4 35                             LDY   YSAV1
FDFF| 60                                RTS
FE00|
FE00|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE00|                   ; + Routine : BL1 (BLank 1)
FE00|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE00|
FE00|                   ; Purpose: Monitor Command Processor entry point for CR command
FE00|
FE00| C6 34             BL1             DEC   YSAV
FE02| F09F                              BEQ   XAM8
FE04|
FE04|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE04|                   ; + Routine : BLANK (BLANK)
FE04|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE04|
FE04|                   ; Purpose: Monitor Command Processor entry point for Blank command
FE04|
FE04| CA                BLANK           DEX
FE05| D016                              BNE   SETMDZ
FE07| C9 BA                             CMP   #0BA
FE09| D0BB                              BNE   XAMPM
FE0B|
FE0B|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE0B|                   ; + Routine : STOR (STORe)
FE0B|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE0B|
FE0B|                   ; Purpose: Monitor Command Processor entry point for Store command
FE0B|
FE0B| 85 31             STOR            STA   MODE
FE0D| A5 3E                             LDA   A2L
FE0F| 91 40                             STA   (A3L),Y
FE11| E6 40                             INC   A3L
FE13| D002                              BNE   RTS5
FE15| E6 41                             INC   A3H
FE17| 60                RTS5            RTS
FE18|
FE18|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                 Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

“_33.PICT” 161 KB 2001-08-13 dpi: 300h x 300v pix: 2243h x 3160v

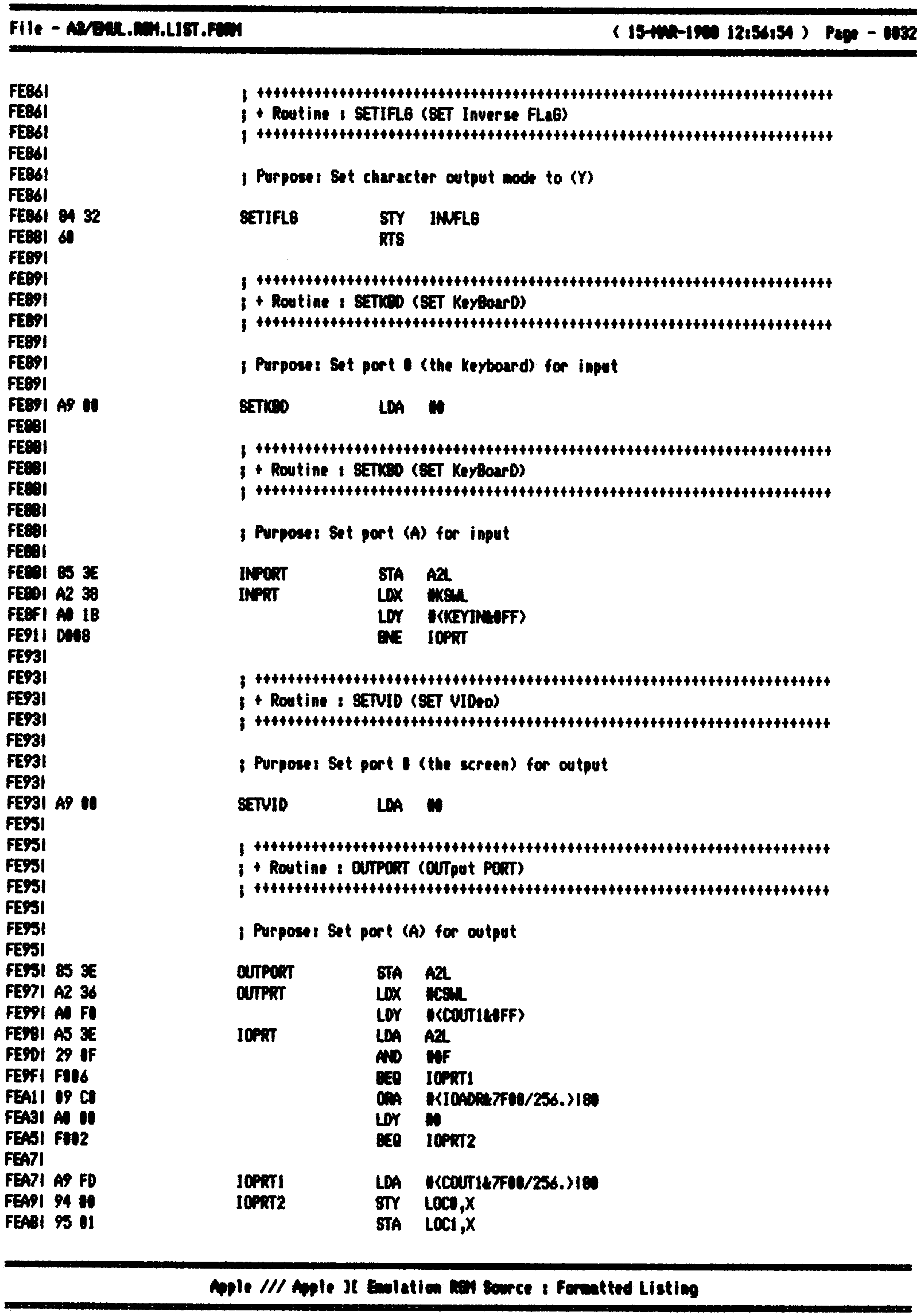


```
FE18|                  ; + Routine : SETMODE (SET monitor MODE)
FE18|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE18|
FE18|                  ; Purpose: Monitor Command Processor, set MODE for colon,
FE18|                  ;          period, plus, or minus
FE18|
FE18| A4 34            SETMODE         LDY   YSAV
FE1A| B9 FF01                          LDA   STACK_TOP,Y
FE1D| 85 31            SETMDZ          STA   MODE
FE1F| 60                               RTS
FE20|
FE20|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE20|                  ; + Routine : LT (Less Than)
FE20|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE20|
FE20|                  ; Purpose: Monitor Command Processor routine for "<" command
FE20|
FE20| A2 01            LT              LDX   #1
FE22| B5 3E            LT2             LDA   A2L,X
FE24| 95 42                            STA   A4L,X
FE26| 95 44                            STA   A5L,X
FE28| CA                               DEX
FE29| 10F7                             BPL   LT2
FE2B| 60                               RTS
FE2C|
FE2C|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE2C|                  ; + Routine : MOVE (MOVE memory)
FE2C|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE2C|
FE2C|                  ; Purpose: Monitor Command Processor routine for Move command.
FE2C|                  ;          (A1L,H) thru (A2L,H) are moved starting at (A4L,H)
FE2C|
FE2C| B1 3C            MOVE            LDA   (A1L),Y
FE2E| 91 42                            STA   (A4L),Y
FE30| 20 B4FC                          JSR   NXTA4
FE33| 90F7                             BCC   MOVE
FE35| 60                               RTS
FE36|
FE36|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE36|                  ; + Routine : VERIFY (VERIFY memory)
FE36|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE36|
FE36|                  ; Purpose: Monitor Command Processor routine for Verify command.
FE36|                  ;          (A1L,H) thru (A2L,H) are compared against (A4L,H)
FE36|
FE36| B1 3C            VERIFY          LDA   (A1L),Y
FE38| D1 42                            CMP   (A4L),Y
FE3A| F01C                             BEQ   VFYOK
FE3C| 20 92FD                          JSR   PRA1
FE3F| B1 3C                            LDA   (A1L),Y
FE41| 20 DAFD                          JSR   PRBYTE
FE44| A9 A0                            LDA   #ASCII_BLANK
FE46| 20 EDFD                          JSR   COUT
FE49| A9 A8                            LDA   #0A8 ; '('
FE4B| 20 EDFD                          JSR   COUT
```



```
FE4E| B1 42                            LDA   (A4L),Y
FE50| 20 DAFD                          JSR   PRBYTE
FE53| A9 A9                            LDA   #0A9 ; ')'
FE55| 20 EDFD                          JSR   COUT
FE58| 20 B4FC          VFYOK           JSR   NXTA4
FE5B| 90D9                             BCC   VERIFY
FE5D| 60                               RTS
FE5E|
FE5E|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE5E|                  ; + Routine : LIST (LIST 6502 instructions)
FE5E|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE5E|
FE5E|                  ; Purpose: Monitor Command Processor routine for List command.
FE5E|                  ;          List 20 instructions thru COUT starting at (PCL,H)
FE5E|
FE5E| 0014             disasm_count    .EQU  20.             ; # lines to disassemble
FE5E|
FE5E| 20 75FE          LIST            JSR   A1PC
FE61| A9 14                            LDA   #disasm_count
FE63| 48               LIST2           PHA
FE64| 20 D0F8                          JSR   INSTDSP
FE67| 20 53F9                          JSR   PCADJ
FE6A| 85 3A                            STA   PCL
FE6C| 84 3B                            STY   PCH
FE6E| 68                               PLA
FE6F| 38                               SEC
FE70| E9 01                            SBC   #1
FE72| D0EF                             BNE   LIST2
FE74| 60                               RTS
FE75|
FE75| 8A               A1PC            TXA
FE76| F007                             BEQ   A1PCRTS
FE78| B5 3C            A1PCLP          LDA   A1L,X
FE7A| 95 3A                            STA   PCL,X
FE7C| CA                               DEX
FE7D| 10F9                             BPL   A1PCLP
FE7F| 60               A1PCRTS         RTS
FE80|
FE80|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE80|                  ; + Routine : SETINV (SET INVerse)
FE80|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE80|
FE80|                  ; Purpose: Set character output mode to Inverse
FE80|
FE80| A0 3F            SETINV          LDY   #3F
FE82| D002                             BNE   SETIFLG
FE84|
FE84|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE84|                  ; + Routine : SETNORM (SET NORMal)
FE84|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE84|
FE84|                  ; Purpose: Set character output mode to Normal
FE84|
FE84| A0 FF            SETNORM         LDY   #0FF
FE86|
```

“_35.PICT” 152 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3160v



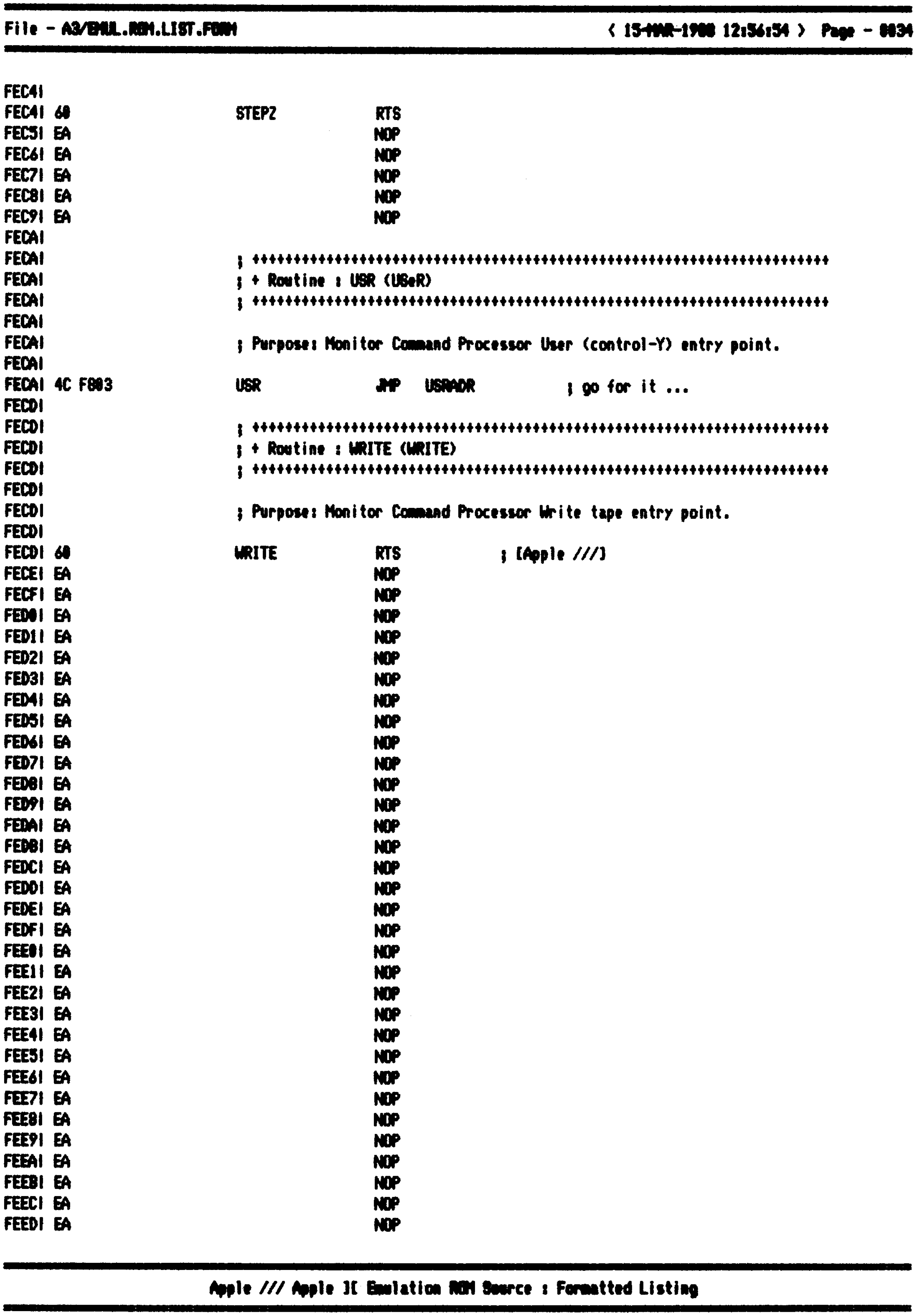
File - A2/EMUL.ROM.LIST.FORM (15-MAR-1980 12:56:54) Page - 0032

```
FE86|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE86|                  ; + Routine : SETIFLG (SET Inverse FLaG)
FE86|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE86|
FE86|                  ; Purpose: Set character output mode to (Y)
FE86|
FE86| 84 32            SETIFLG      STY   INVFLG
FE88| 60                            RTS
FE89|
FE89|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE89|                  ; + Routine : SETKBD (SET KeyBoarD)
FE89|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE89|
FE89|                  ; Purpose: Set port 0 (the keyboard) for input
FE89|
FE89| A9 00            SETKBD       LDA   #0
FE8B|
FE8B|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE8B|                  ; + Routine : SETKBD (SET KeyBoarD)
FE8B|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE8B|
FE8B|                  ; Purpose: Set port (A) for input
FE8B|
FE8B| 85 3E            INPORT       STA   A2L
FE8D| A2 38            INPRT        LDX   #KSWL
FE8F| A0 1B                         LDY   #(KEYIN&0FF)
FE91| D008                          BNE   IOPRT
FE93|
FE93|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE93|                  ; + Routine : SETVID (SET VIDeo)
FE93|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE93|
FE93|                  ; Purpose: Set port 0 (the screen) for output
FE93|
FE93| A9 00            SETVID       LDA   #0
FE95|
FE95|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE95|                  ; + Routine : OUTPORT (OUTput PORT)
FE95|                  ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FE95|
FE95|                  ; Purpose: Set port (A) for output
FE95|
FE95| 85 3E            OUTPORT      STA   A2L
FE97| A2 36            OUTPRT       LDX   #CSWL
FE99| A0 F0                         LDY   #(COUT1&0FF)
FE9B| A5 3E            IOPRT        LDA   A2L
FE9D| 29 0F                         AND   #0F
FE9F| F006                          BEQ   IOPRT1
FEA1| 09 C0                         ORA   #(IOADR&7F00/256.)!80
FEA3| A0 00                         LDY   #0
FEA5| F002                          BEQ   IOPRT2
FEA7|
FEA7| A9 FD            IOPRT1       LDA   #(COUT1&7F00/256.)!80
FEA9| 94 00            IOPRT2       STY   LOC0,X
FEAB| 95 01                         STA   LOC1,X
```

Apple /// Apple][Emulation ROM Source : Formatted Listing

“_36.PICT” 166 KB 2001-08-13 dpi: 300h x 300v pix: 2225h x 3172v

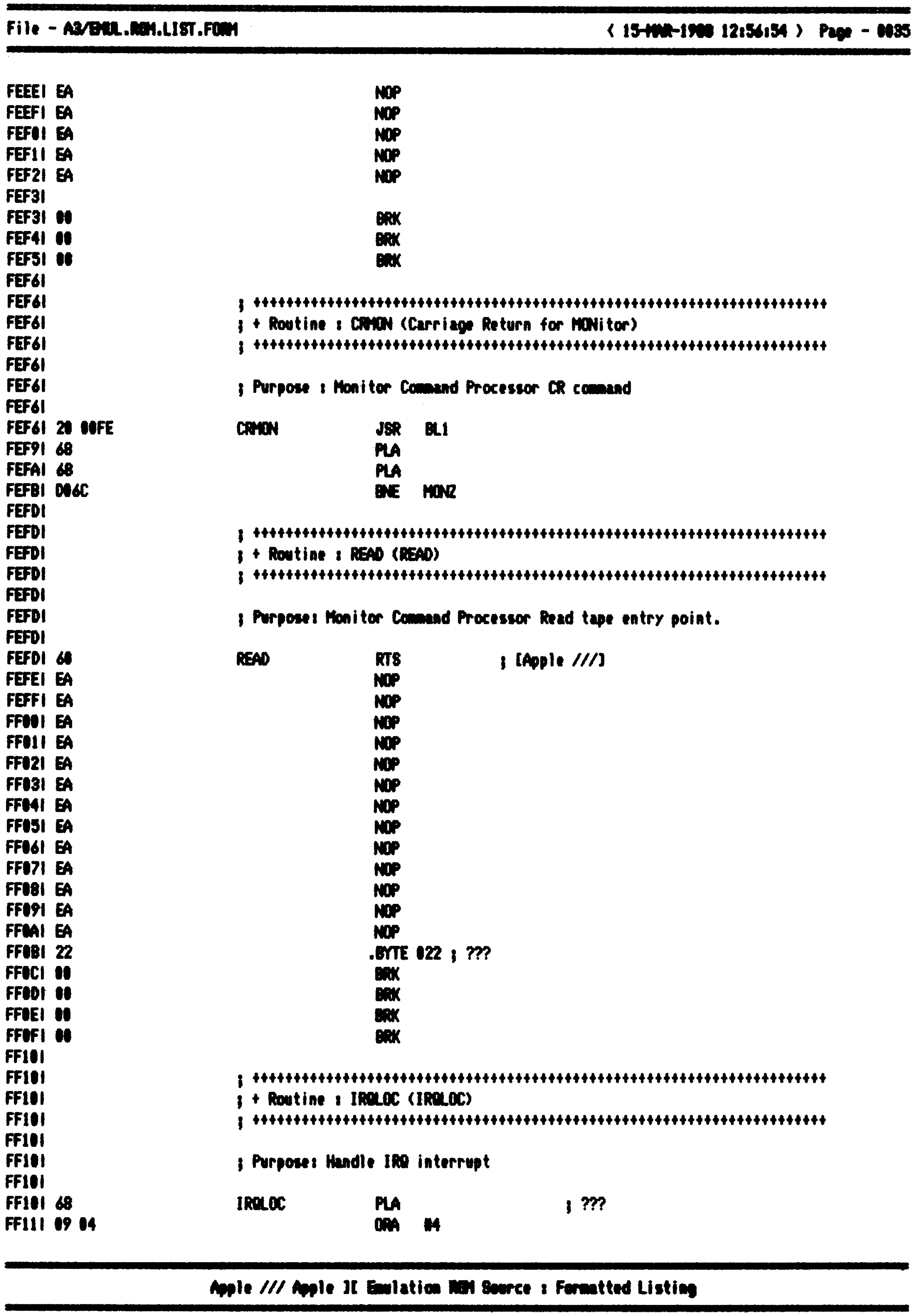
```
FEAD| 60                              RTS
FEAE|
FEAE| EA                              NOP
FEAF| EA                              NOP
FEB0|
FEB0|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEB0|                     ; + Routine : XBASIC (eXecute BASIC)
FEB0|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEB0|
FEB0|                     ; Purpose: Execute the BASIC language (cold boot)
FEB0|
FEB0| 4C 00E0             XBASIC      JMP   BASIC
FEB3|
FEB3|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEB3|                     ; + Routine : BASCONT (BASic CONTinue)
FEB3|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEB3|
FEB3|                     ; Purpose: Execute the BASIC language (warm boot)
FEB3|
FEB3| 4C 03E0             BASCONT     JMP   BASIC2
FEB6|
FEB6|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEB6|                     ; + Routine : GO (GO to routine)
FEB6|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEB6|
FEB6|                     ; Purpose: Monitor Command Processor Go entry point.
FEB6|                     ;          Set PCL,H from A1L,H, restore all registers, and
FEB6|                     ;          begin code execution starting at PCL,H.
FEB6|
FEB6| 20 75FE             GO          JSR   A1PC
FEB9| 20 3FFF                         JSR   RESTORE
FEBC| 6C 3A00                         JMP   @PCL
FEBF|
FEBF|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEBF|                     ; + Routine : REGZ (REGister diaplay Z)
FEBF|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEBF|
FEBF|                     ; Purpose: Monitor Command Processor Display Registers entry point.
FEBF|
FEBF| 4C D7FA             REGZ        JMP   REGDSP
FEC2|
FEC2|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEC2|                     ; + Routine : TRACE (TRACE)
FEC2|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEC2|
FEC2|                     ; Purpose: Monitor Command Processor Trace entry point.
FEC2|
FEC2| 60                  TRACE       RTS
FEC3| EA                              NOP
FEC4|
FEC4|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEC4|                     ; + Routine : STEPZ (STEP Z)
FEC4|                     ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEC4|
FEC4|                     ; Purpose: Monitor Command Processor Step entry point.
```



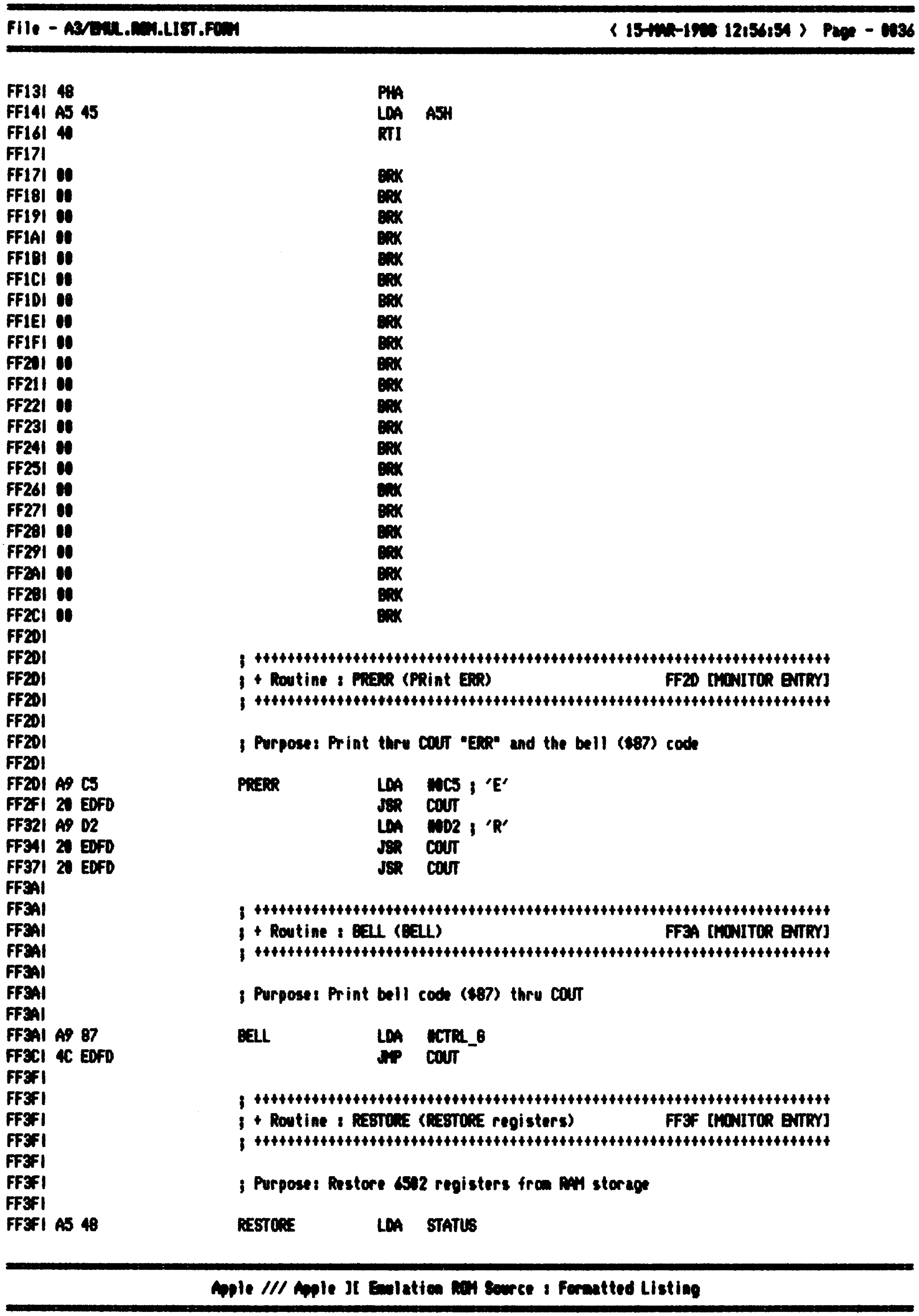
```
File - A3/EMUL.ROM.LIST.FORM                          < 15-MAR-1980 12:56:54 >  Page - 0034

FEC4|
FEC4| 60                STEPZ         RTS
FEC5| EA                              NOP
FEC6| EA                              NOP
FEC7| EA                              NOP
FEC8| EA                              NOP
FEC9| EA                              NOP
FECA|
FECA|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FECA|                   ; + Routine : USR (USeR)
FECA|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FECA|
FECA|                   ; Purpose: Monitor Command Processor User (control-Y) entry point.
FECA|
FECA| 4C F803           USR           JMP   USRADR         ; go for it ...
FECD|
FECD|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FECD|                   ; + Routine : WRITE (WRITE)
FECD|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FECD|
FECD|                   ; Purpose: Monitor Command Processor Write tape entry point.
FECD|
FECD| 60                WRITE         RTS            ; [Apple ///]
FECE| EA                              NOP
FECF| EA                              NOP
FED0| EA                              NOP
FED1| EA                              NOP
FED2| EA                              NOP
FED3| EA                              NOP
FED4| EA                              NOP
FED5| EA                              NOP
FED6| EA                              NOP
FED7| EA                              NOP
FED8| EA                              NOP
FED9| EA                              NOP
FEDA| EA                              NOP
FEDB| EA                              NOP
FEDC| EA                              NOP
FEDD| EA                              NOP
FEDE| EA                              NOP
FEDF| EA                              NOP
FEE0| EA                              NOP
FEE1| EA                              NOP
FEE2| EA                              NOP
FEE3| EA                              NOP
FEE4| EA                              NOP
FEE5| EA                              NOP
FEE6| EA                              NOP
FEE7| EA                              NOP
FEE8| EA                              NOP
FEE9| EA                              NOP
FEEA| EA                              NOP
FEEB| EA                              NOP
FEEC| EA                              NOP
FEED| EA                              NOP
```

Apple /// Apple][Emulation ROM Source : Formatted Listing

"_38.PICT" 119 KB 2001-08-13 dpi: 300h x 300v pix: 2243h x 3172v

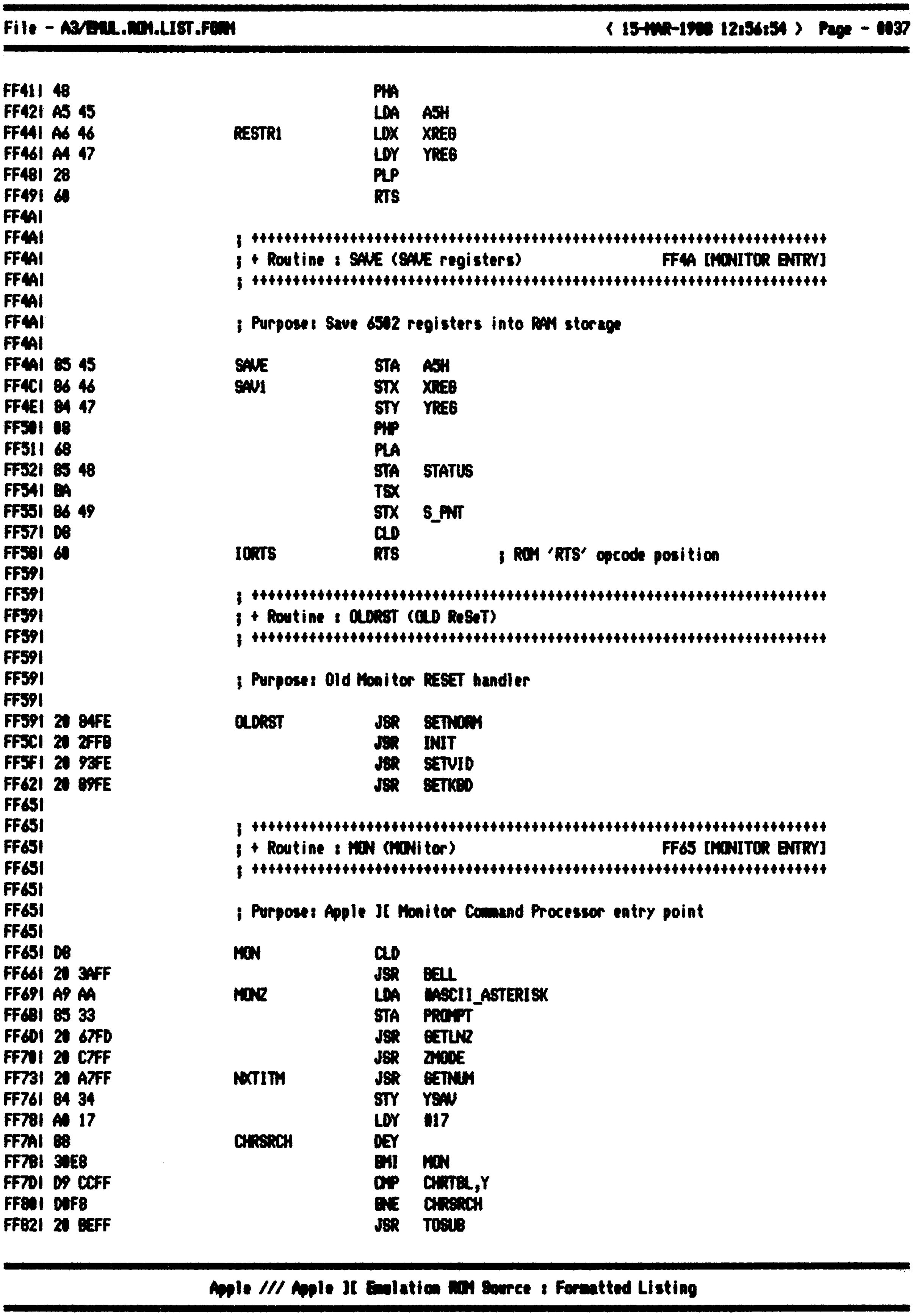


```
FEEE| EA                         NOP
FEEF| EA                         NOP
FEF0| EA                         NOP
FEF1| EA                         NOP
FEF2| EA                         NOP
FEF3|
FEF3| 00                         BRK
FEF4| 00                         BRK
FEF5| 00                         BRK
FEF6|
FEF6|               ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEF6|               ; + Routine : CRMON (Carriage Return for MONitor)
FEF6|               ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEF6|
FEF6|               ; Purpose : Monitor Command Processor CR command
FEF6|
FEF6| 20 00FE       CRMON        JSR   BL1
FEF9| 68                         PLA
FEFA| 68                         PLA
FEFB| D06C                       BNE   MONZ
FEFD|
FEFD|               ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEFD|               ; + Routine : READ (READ)
FEFD|               ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FEFD|
FEFD|               ; Purpose: Monitor Command Processor Read tape entry point.
FEFD|
FEFD| 60            READ         RTS          ; [Apple ///]
FEFE| EA                         NOP
FEFF| EA                         NOP
FF00| EA                         NOP
FF01| EA                         NOP
FF02| EA                         NOP
FF03| EA                         NOP
FF04| EA                         NOP
FF05| EA                         NOP
FF06| EA                         NOP
FF07| EA                         NOP
FF08| EA                         NOP
FF09| EA                         NOP
FF0A| EA                         NOP
FF0B| 22                        .BYTE 022 ; ???
FF0C| 00                         BRK
FF0D| 00                         BRK
FF0E| 00                         BRK
FF0F| 00                         BRK
FF10|
FF10|               ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF10|               ; + Routine : IRQLOC (IRQLOC)
FF10|               ; +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF10|
FF10|               ; Purpose: Handle IRQ interrupt
FF10|
FF10| 68            IRQLOC       PLA                 ; ???
FF11| 09 04                      ORA   #4
```

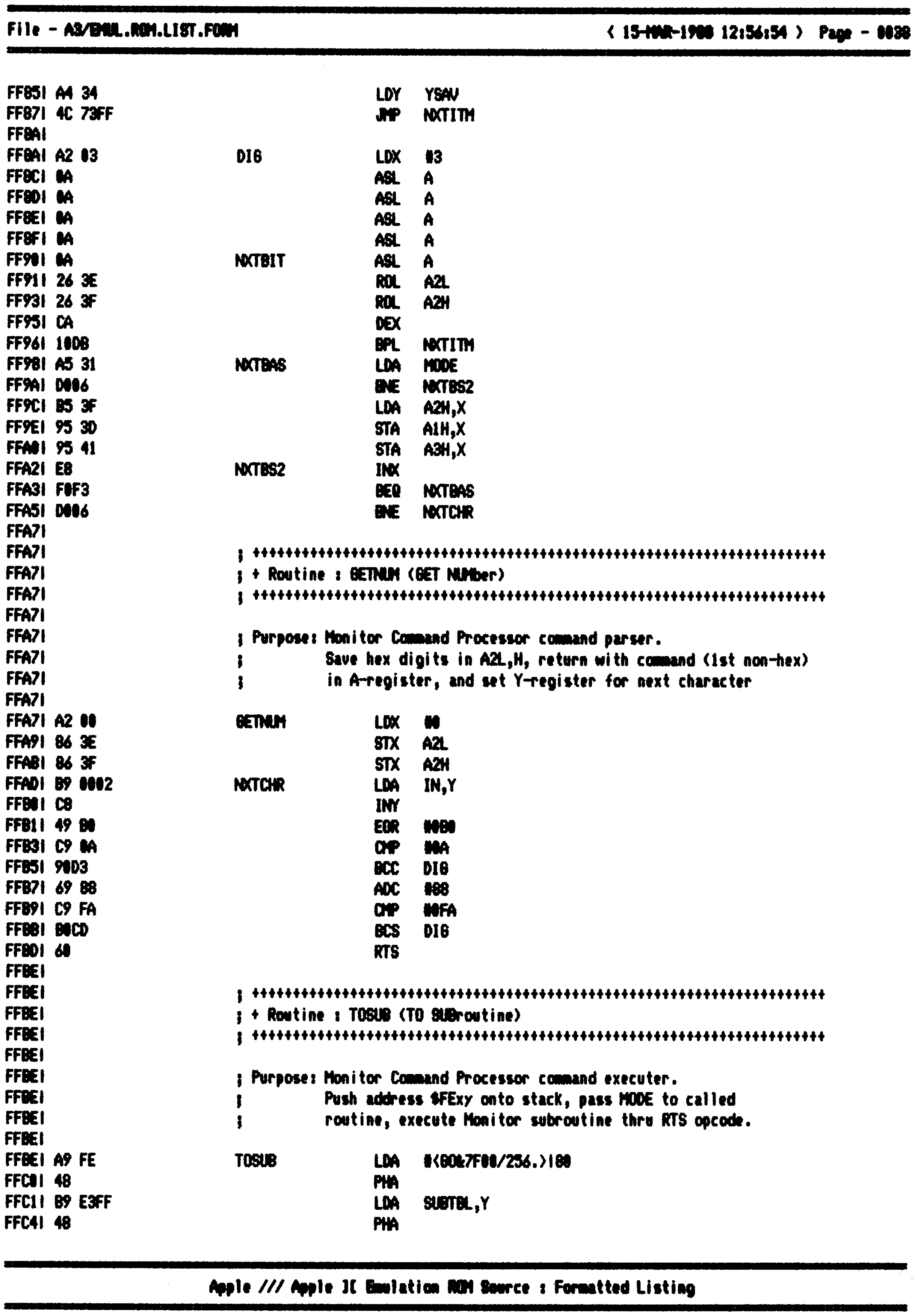


```
FF13: 48                         PHA
FF14: A5 45                      LDA   A5H
FF16: 40                         RTI
FF17:
FF17: 00                         BRK
FF18: 00                         BRK
FF19: 00                         BRK
FF1A: 00                         BRK
FF1B: 00                         BRK
FF1C: 00                         BRK
FF1D: 00                         BRK
FF1E: 00                         BRK
FF1F: 00                         BRK
FF20: 00                         BRK
FF21: 00                         BRK
FF22: 00                         BRK
FF23: 00                         BRK
FF24: 00                         BRK
FF25: 00                         BRK
FF26: 00                         BRK
FF27: 00                         BRK
FF28: 00                         BRK
FF29: 00                         BRK
FF2A: 00                         BRK
FF2B: 00                         BRK
FF2C: 00                         BRK
FF2D:
FF2D:               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF2D:               ; + Routine : PRERR (PRint ERR)                     FF2D [MONITOR ENTRY]
FF2D:               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF2D:
FF2D:               ; Purpose: Print thru COUT "ERR" and the bell ($87) code
FF2D:
FF2D: A9 C5         PRERR        LDA   #$C5 ; 'E'
FF2F: 20 EDFD                    JSR   COUT
FF32: A9 D2                      LDA   #$D2 ; 'R'
FF34: 20 EDFD                    JSR   COUT
FF37: 20 EDFD                    JSR   COUT
FF3A:
FF3A:               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF3A:               ; + Routine : BELL (BELL)                           FF3A [MONITOR ENTRY]
FF3A:               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF3A:
FF3A:               ; Purpose: Print bell code ($87) thru COUT
FF3A:
FF3A: A9 87         BELL         LDA   #CTRL_G
FF3C: 4C EDFD                    JMP   COUT
FF3F:
FF3F:               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF3F:               ; + Routine : RESTORE (RESTORE registers)           FF3F [MONITOR ENTRY]
FF3F:               ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF3F:
FF3F:               ; Purpose: Restore 6502 registers from RAM storage
FF3F:
FF3F: A5 48         RESTORE      LDA   STATUS
```

“_40.PICT” 139 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3160v

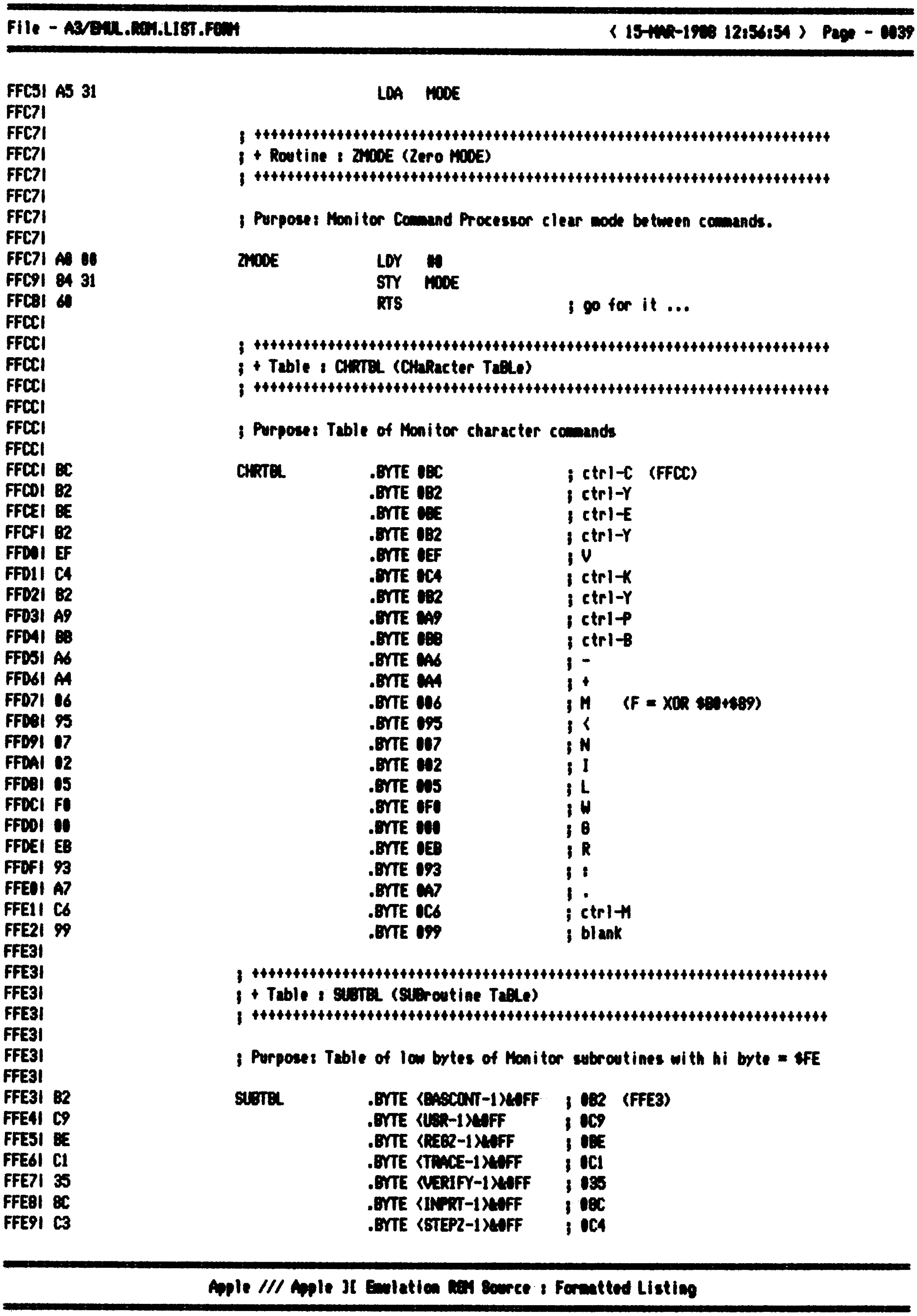


```
FF41| 48                              PHA
FF42| A5 45                           LDA   A5H
FF44| A6 46             RESTR1        LDX   XREG
FF46| A4 47                           LDY   YREG
FF48| 28                              PLP
FF49| 60                              RTS
FF4A|
FF4A|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF4A|                   ; + Routine : SAVE (SAVE registers)                FF4A [MONITOR ENTRY]
FF4A|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF4A|
FF4A|                   ; Purpose: Save 6502 registers into RAM storage
FF4A|
FF4A| 85 45             SAVE          STA   A5H
FF4C| 86 46             SAV1          STX   XREG
FF4E| 84 47                           STY   YREG
FF50| 08                              PHP
FF51| 68                              PLA
FF52| 85 48                           STA   STATUS
FF54| BA                              TSX
FF55| 86 49                           STX   S_PNT
FF57| D8                              CLD
FF58| 60                IORTS         RTS             ; ROM 'RTS' opcode position
FF59|
FF59|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF59|                   ; + Routine : OLDRST (OLD ReSeT)
FF59|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF59|
FF59|                   ; Purpose: Old Monitor RESET handler
FF59|
FF59| 20 84FE           OLDRST        JSR   SETNORM
FF5C| 20 2FFB                         JSR   INIT
FF5F| 20 93FE                         JSR   SETVID
FF62| 20 89FE                         JSR   SETKBD
FF65|
FF65|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF65|                   ; + Routine : MON (MONitor)                        FF65 [MONITOR ENTRY]
FF65|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FF65|
FF65|                   ; Purpose: Apple ][ Monitor Command Processor entry point
FF65|
FF65| D8                MON           CLD
FF66| 20 3AFF                         JSR   BELL
FF69| A9 AA             MONZ          LDA   #ASCII_ASTERISK
FF6B| 85 33                           STA   PROMPT
FF6D| 20 67FD                         JSR   GETLNZ
FF70| 20 C7FF                         JSR   ZMODE
FF73| 20 A7FF           NXTITM        JSR   GETNUM
FF76| 84 34                           STY   YSAV
FF78| A0 17                           LDY   #17
FF7A| 88                CHRSRCH       DEY
FF7B| 30E8                            BMI   MON
FF7D| D9 CCFF                         CMP   CHRTBL,Y
FF80| D0F8                            BNE   CHRSRCH
FF82| 20 BEFF                         JSR   TOSUB
```



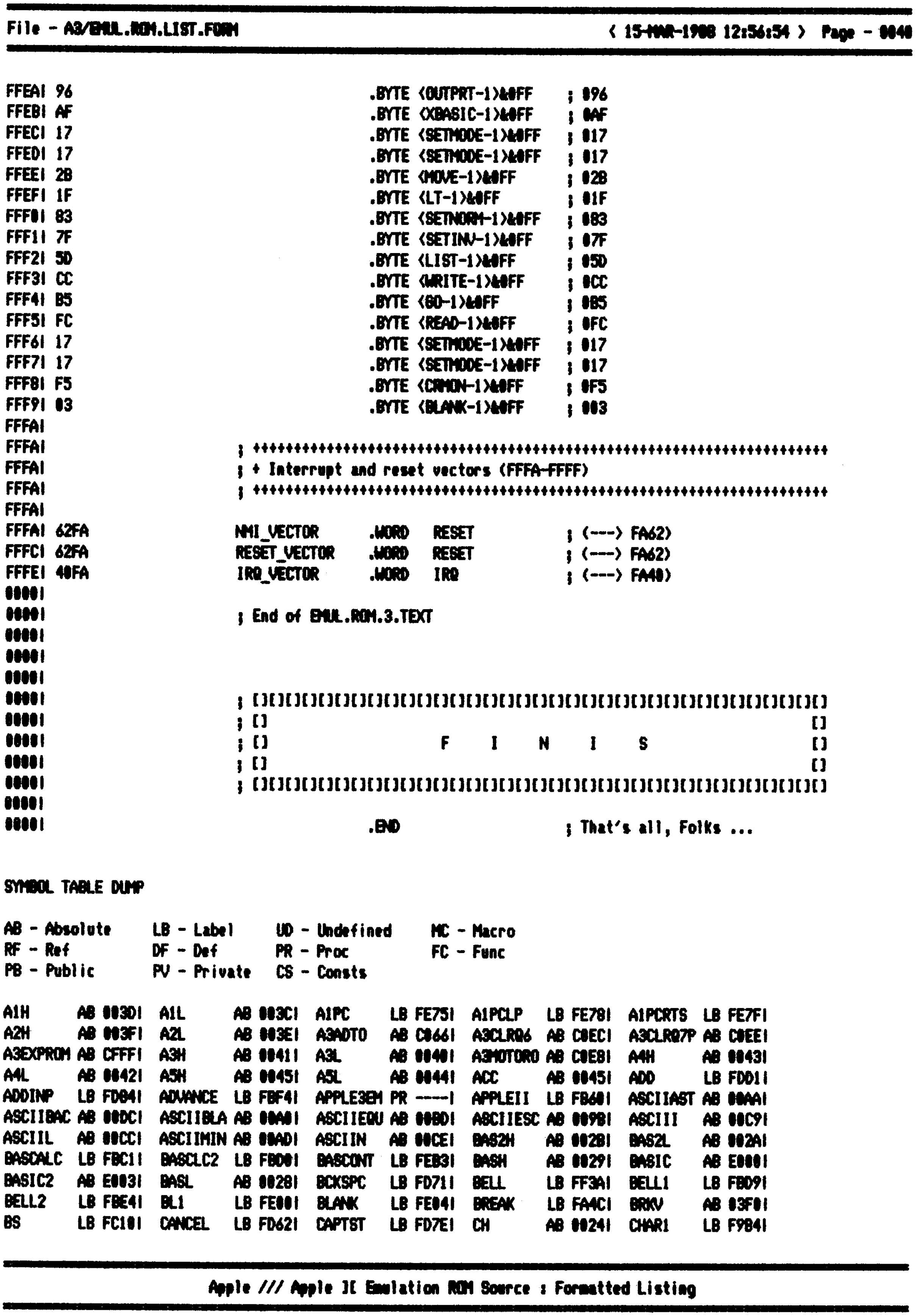
```
FF85| A4 34                         LDY   YSAV
FF87| 4C 73FF                       JMP   NXTITM
FF8A|
FF8A| A2 03          DIG            LDX   #3
FF8C| 0A                            ASL   A
FF8D| 0A                            ASL   A
FF8E| 0A                            ASL   A
FF8F| 0A                            ASL   A
FF90| 0A             NXTBIT         ASL   A
FF91| 26 3E                         ROL   A2L
FF93| 26 3F                         ROL   A2H
FF95| CA                            DEX
FF96| 10DB                          BPL   NXTITM
FF98| A5 31          NXTBAS         LDA   MODE
FF9A| D006                          BNE   NXTBS2
FF9C| B5 3F                         LDA   A2H,X
FF9E| 95 3D                         STA   A1H,X
FFA0| 95 41                         STA   A3H,X
FFA2| E8             NXTBS2         INX
FFA3| F0F3                          BEQ   NXTBAS
FFA5| D006                          BNE   NXTCHR
FFA7|
FFA7|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFA7|                ; + Routine : GETNUM (GET NUMber)
FFA7|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFA7|
FFA7|                ; Purpose: Monitor Command Processor command parser.
FFA7|                ;          Save hex digits in A2L,H, return with command (1st non-hex)
FFA7|                ;          in A-register, and set Y-register for next character
FFA7|
FFA7| A2 00          GETNUM         LDX   #0
FFA9| 86 3E                         STX   A2L
FFAB| 86 3F                         STX   A2H
FFAD| B9 0002        NXTCHR         LDA   IN,Y
FFB0| C8                            INY
FFB1| 49 B0                         EOR   #0B0
FFB3| C9 0A                         CMP   #0A
FFB5| 90D3                          BCC   DIG
FFB7| 69 88                         ADC   #88
FFB9| C9 FA                         CMP   #0FA
FFBB| B0CD                          BCS   DIG
FFBD| 60                            RTS
FFBE|
FFBE|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFBE|                ; + Routine : TOSUB (TO SUBroutine)
FFBE|                ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFBE|
FFBE|                ; Purpose: Monitor Command Processor command executer.
FFBE|                ;          Push address $FExy onto stack, pass MODE to called
FFBE|                ;          routine, execute Monitor subroutine thru RTS opcode.
FFBE|
FFBE| A9 FE          TOSUB          LDA   #<GO&7F00/256.>!00
FFC0| 48                            PHA
FFC1| B9 E3FF                       LDA   SUBTBL,Y
FFC4| 48                            PHA
```

“_42.PICT” 149 KB 2001-08-13 dpi: 300h x 300v pix: 2219h x 3167v



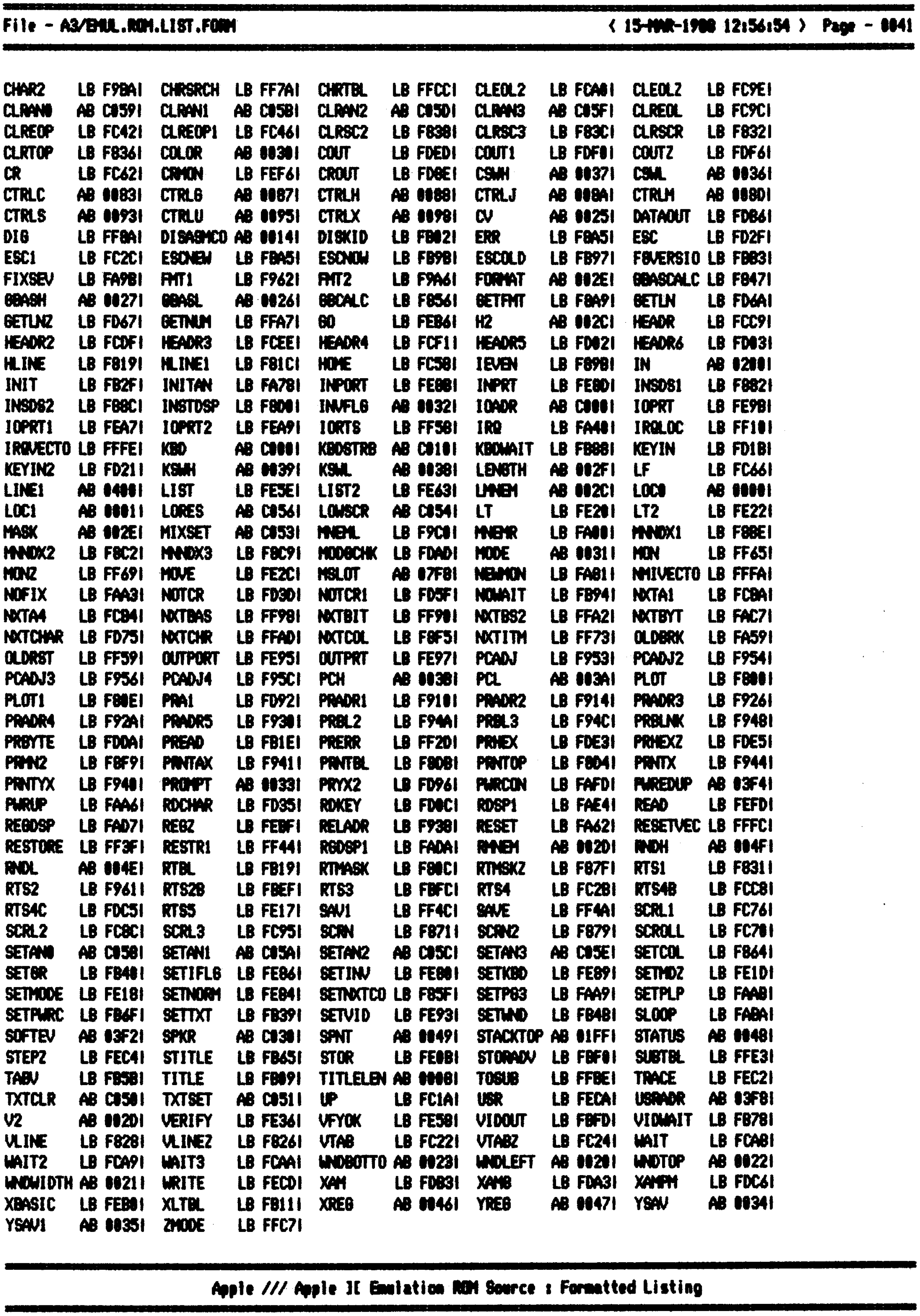
```
File - A3/EMUL.ROM.LIST.FORM                                   ( 15-MAR-1988 12:56:54 )  Page - 0039

FFC5| A5 31                          LDA   MODE
FFC7|
FFC7|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFC7|                   ; + Routine : ZMODE (Zero MODE)
FFC7|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFC7|
FFC7|                   ; Purpose: Monitor Command Processor clear mode between commands.
FFC7|
FFC7| A0 00             ZMODE        LDY   #0
FFC9| 84 31                          STY   MODE
FFCB| 60                             RTS                  ; go for it ...
FFCC|
FFCC|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFCC|                   ; + Table : CHRTBL (CHaRacter TaBLe)
FFCC|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFCC|
FFCC|                   ; Purpose: Table of Monitor character commands
FFCC|
FFCC| BC                CHRTBL      .BYTE 0BC             ; ctrl-C  (FFCC)
FFCD| B2                            .BYTE 0B2             ; ctrl-Y
FFCE| BE                            .BYTE 0BE             ; ctrl-E
FFCF| B2                            .BYTE 0B2             ; ctrl-Y
FFD0| EF                            .BYTE 0EF             ; V
FFD1| C4                            .BYTE 0C4             ; ctrl-K
FFD2| B2                            .BYTE 0B2             ; ctrl-Y
FFD3| A9                            .BYTE 0A9             ; ctrl-P
FFD4| BB                            .BYTE 0BB             ; ctrl-B
FFD5| A6                            .BYTE 0A6             ; -
FFD6| A4                            .BYTE 0A4             ; +
FFD7| 06                            .BYTE 006             ; M    (F = XOR $B0+$89)
FFD8| 95                            .BYTE 095             ; <
FFD9| 07                            .BYTE 007             ; N
FFDA| 02                            .BYTE 002             ; I
FFDB| 05                            .BYTE 005             ; L
FFDC| F0                            .BYTE 0F0             ; W
FFDD| 00                            .BYTE 000             ; G
FFDE| EB                            .BYTE 0EB             ; R
FFDF| 93                            .BYTE 093             ; :
FFE0| A7                            .BYTE 0A7             ; .
FFE1| C6                            .BYTE 0C6             ; ctrl-M
FFE2| 99                            .BYTE 099             ; blank
FFE3|
FFE3|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFE3|                   ; + Table : SUBTBL (SUBroutine TaBLe)
FFE3|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFE3|
FFE3|                   ; Purpose: Table of low bytes of Monitor subroutines with hi byte = $FE
FFE3|
FFE3| B2                SUBTBL      .BYTE (BASCONT-1)&0FF  ; 0B2  (FFE3)
FFE4| C9                            .BYTE (USR-1)&0FF      ; 0C9
FFE5| BE                            .BYTE (REGZ-1)&0FF     ; 0BE
FFE6| C1                            .BYTE (TRACE-1)&0FF    ; 0C1
FFE7| 35                            .BYTE (VERIFY-1)&0FF   ; 035
FFE8| 8C                            .BYTE (INPRT-1)&0FF    ; 08C
FFE9| C3                            .BYTE (STEPZ-1)&0FF    ; 0C4

                  Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

"_43.PICT" 164 KB 2001-08-13 dpi: 300h x 300v pix: 2237h x 3172v



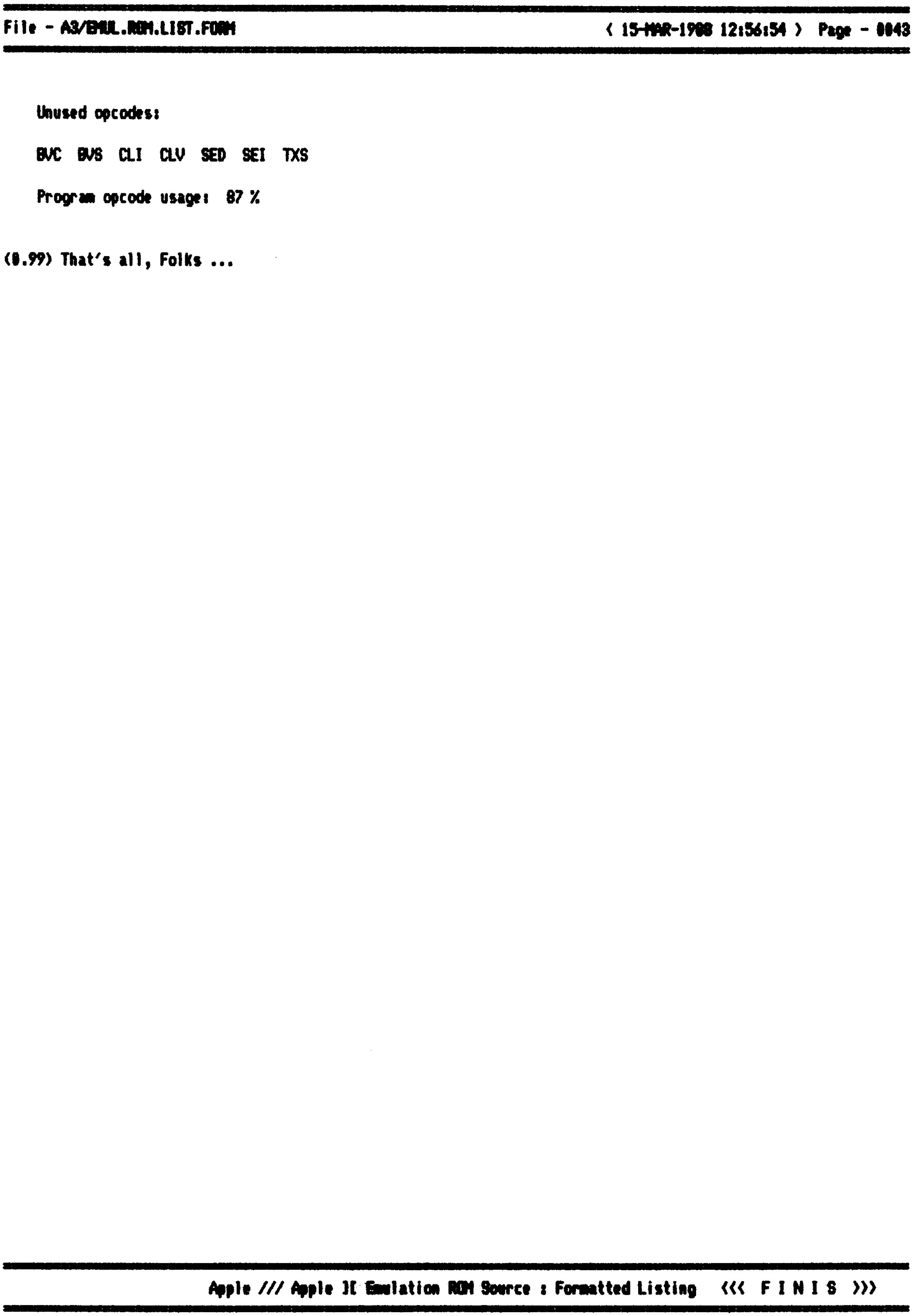
```
FFEA| 96                            .BYTE <OUTPRT-1>&0FF     ; 096
FFEB| AF                            .BYTE <XBASIC-1>&0FF     ; 0AF
FFEC| 17                            .BYTE <SETMODE-1>&0FF    ; 017
FFED| 17                            .BYTE <SETMODE-1>&0FF    ; 017
FFEE| 2B                            .BYTE <MOVE-1>&0FF       ; 02B
FFEF| 1F                            .BYTE <LT-1>&0FF         ; 01F
FFF0| 83                            .BYTE <SETNORM-1>&0FF    ; 083
FFF1| 7F                            .BYTE <SETINV-1>&0FF     ; 07F
FFF2| 5D                            .BYTE <LIST-1>&0FF       ; 05D
FFF3| CC                            .BYTE <WRITE-1>&0FF      ; 0CC
FFF4| B5                            .BYTE <GO-1>&0FF         ; 0B5
FFF5| FC                            .BYTE <READ-1>&0FF       ; 0FC
FFF6| 17                            .BYTE <SETMODE-1>&0FF    ; 017
FFF7| 17                            .BYTE <SETMODE-1>&0FF    ; 017
FFF8| F5                            .BYTE <CRMON-1>&0FF      ; 0F5
FFF9| 03                            .BYTE <BLANK-1>&0FF      ; 003
FFFA|
FFFA|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFFA|                   ; + Interrupt and reset vectors (FFFA-FFFF)
FFFA|                   ; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
FFFA|
FFFA| 62FA              NMI_VECTOR      .WORD   RESET            ; (---> FA62)
FFFC| 62FA              RESET_VECTOR    .WORD   RESET            ; (---> FA62)
FFFE| 40FA              IRQ_VECTOR      .WORD   IRQ              ; (---> FA40)
0000|
0000|                   ; End of EMUL.ROM.3.TEXT
0000|
0000|
0000|
0000|                   ; [][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][]
0000|                   ; []                                                                    []
0000|                   ; []                  F     I     N     I     S                         []
0000|                   ; []                                                                    []
0000|                   ; [][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][][]
0000|
0000|                               .END                 ; That's all, Folks ...


SYMBOL TABLE DUMP

AB - Absolute    LB - Label      UD - Undefined   MC - Macro
RF - Ref         DF - Def        PR - Proc        FC - Func
PB - Public      PV - Private    CS - Consts

A1H      AB 003D| A1L      AB 003C| A1PC     LB FE75| A1PCLP   LB FE78| A1PCRTS  LB FE7F|
A2H      AB 003F| A2L      AB 003E| A3ADTO   AB C066| A3CLRQ6  AB C0EC| A3CLRQ7P AB C0EE|
A3EXPROM AB CFFF| A3H      AB 0041| A3L      AB 0040| A3MOTORO AB C0E8| A4H      AB 0043|
A4L      AB 0042| A5H      AB 0045| A5L      AB 0044| ACC      AB 0045| ADD      LB FDD1|
ADDINP   LB FD84| ADVANCE  LB FBF4| APPLE3EM PR ----| APPLEII  LB FB60| ASCIIAST AB 00AA|
ASCIIBAC AB 00DC| ASCIIBLA AB 00A0| ASCIIEQU AB 00BD| ASCIIESC AB 009B| ASCIII   AB 00C9|
ASCIIL   AB 00CC| ASCIIMIN AB 00AD| ASCIIN   AB 00CE| BAS2H    AB 002B| BAS2L    AB 002A|
BASCALC  LB FBC1| BASCLC2  LB FBD0| BASCONT  LB FEB3| BASH     AB 0029| BASIC    AB E000|
BASIC2   AB E003| BASL     AB 0028| BCKSPC   LB FD71| BELL     LB FF3A| BELL1    LB FBD9|
BELL2    LB FBE4| BL1      LB FE00| BLANK    LB FE04| BREAK    LB FA4C| BRKV     AB 03F0|
BS       LB FC10| CANCEL   LB FD62| CAPTST   LB FD7E| CH       AB 0024| CHAR1    LB F9B4|
```

"_44.PICT" 204 KB 2001-08-13 dpi: 300h x 300v pix: 2231h x 3190v



```
File - A3/EMUL.ROM.LIST.FORM                                          ( 15-MAR-1988 12:56:54 )  Page - 0041

CHAR2    LB F9BA|  CHRSRCH  LB FF7A|  CHRTBL   LB FFCC|  CLEOL2   LB FCA0|  CLEOLZ   LB FC9E|
CLRAN0   AB C059|  CLRAN1   AB C05B|  CLRAN2   AB C05D|  CLRAN3   AB C05F|  CLREOL   LB FC9C|
CLREOP   LB FC42|  CLREOP1  LB FC46|  CLRSC2   LB F838|  CLRSC3   LB F83C|  CLRSCR   LB F832|
CLRTOP   LB F836|  COLOR    AB 0030|  COUT     LB FDED|  COUT1    LB FDF0|  COUTZ    LB FDF6|
CR       LB FC62|  CRMON    LB FEF6|  CROUT    LB FD8E|  CSWH     AB 0037|  CSWL     AB 0036|
CTRLC    AB 0083|  CTRLG    AB 0087|  CTRLH    AB 0088|  CTRLJ    AB 008A|  CTRLM    AB 008D|
CTRLS    AB 0093|  CTRLU    AB 0095|  CTRLX    AB 0098|  CV       AB 0025|  DATAOUT  LB FDB6|
DIG      LB FF8A|  DISASMCO AB 0014|  DISKID   LB FB02|  ERR      LB F8A5|  ESC      LB FD2F|
ESC1     LB FC2C|  ESCNEW   LB FBA5|  ESCNOW   LB FB9B|  ESCOLD   LB FB97|  F8VERSIO LB FB03|
FIXSEV   LB FA9B|  FMT1     LB F962|  FMT2     LB F9A6|  FORMAT   AB 002E|  GBASCALC LB F847|
GBASH    AB 0027|  GBASL    AB 0026|  GBCALC   LB F856|  GETFMT   LB F8A9|  GETLN    LB FD6A|
GETLNZ   LB FD67|  GETNUM   LB FFA7|  GO       LB FEB6|  H2       AB 002C|  HEADR    LB FCC9|
HEADR2   LB FCDF|  HEADR3   LB FCEE|  HEADR4   LB FCF1|  HEADR5   LB FD02|  HEADR6   LB FD03|
HLINE    LB F819|  HLINE1   LB F81C|  HOME     LB FC58|  IEVEN    LB F89B|  IN       AB 0200|
INIT     LB FB2F|  INITAN   LB FA78|  INPORT   LB FE8B|  INPRT    LB FE8D|  INSDS1   LB F882|
INSDS2   LB F88C|  INSTDSP  LB F8D0|  INVFLG   AB 0032|  IOADR    AB C000|  IOPRT    LB FE9B|
IOPRT1   LB FEA7|  IOPRT2   LB FEA9|  IORTS    LB FF58|  IRQ      LB FA40|  IRQLOC   LB FF10|
IRQVECTO LB FFFE|  KBD      AB C000|  KBDSTRB  AB C010|  KBDWAIT  LB FB88|  KEYIN    LB FD1B|
KEYIN2   LB FD21|  KSWH     AB 0039|  KSWL     AB 0038|  LENGTH   AB 002F|  LF       LB FC66|
LINE1    AB 0400|  LIST     LB FE5E|  LIST2    LB FE63|  LMNEM    AB 002C|  LOC0     AB 0000|
LOC1     AB 0001|  LORES    AB C056|  LOWSCR   AB C054|  LT       LB FE20|  LT2      LB FE22|
MASK     AB 002E|  MIXSET   AB C053|  MNEML    LB F9C0|  MNEMR    LB FA00|  MNNDX1   LB F8BE|
MNNDX2   LB F8C2|  MNNDX3   LB F8C9|  MODGCHK  LB FDAD|  MODE     AB 0031|  MON      LB FF65|
MONZ     LB FF69|  MOVE     LB FE2C|  MSLOT    AB 07F8|  NEWMON   LB FA81|  NMIVECTO LB FFFA|
NOFIX    LB FAA3|  NOTCR    LB FD3D|  NOTCR1   LB FD5F|  NOWAIT   LB FB94|  NXTA1    LB FCBA|
NXTA4    LB FCB4|  NXTBAS   LB FF98|  NXTBIT   LB FF90|  NXTBS2   LB FFA2|  NXTBYT   LB FAC7|
NXTCHAR  LB FD75|  NXTCHR   LB FFAD|  NXTCOL   LB F8F5|  NXTITM   LB FF73|  OLDBRK   LB FA59|
OLDRST   LB FF59|  OUTPORT  LB FE95|  OUTPRT   LB FE97|  PCADJ    LB F953|  PCADJ2   LB F954|
PCADJ3   LB F956|  PCADJ4   LB F95C|  PCH      AB 003B|  PCL      AB 003A|  PLOT     LB F800|
PLOT1    LB F80E|  PRA1     LB FD92|  PRADR1   LB F910|  PRADR2   LB F914|  PRADR3   LB F926|
PRADR4   LB F92A|  PRADR5   LB F930|  PRBL2    LB F94A|  PRBL3    LB F94C|  PRBLNK   LB F948|
PRBYTE   LB FDDA|  PREAD    LB FB1E|  PRERR    LB FF2D|  PRHEX    LB FDE3|  PRHEXZ   LB FDE5|
PRMN2    LB F8F9|  PRNTAX   LB F941|  PRNTBL   LB F8DB|  PRNTOP   LB F8D4|  PRNTX    LB F944|
PRNTYX   LB F940|  PROMPT   AB 0033|  PRYX2    LB FD96|  PWRCON   LB FAFD|  PWREDUP  AB 03F4|
PWRUP    LB FAA6|  RDCHAR   LB FD35|  RDKEY    LB FD0C|  RDSP1    LB FAE4|  READ     LB FEFD|
REGDSP   LB FAD7|  REGZ     LB FEBF|  RELADR   LB F938|  RESET    LB FA62|  RESETVEC LB FFFC|
RESTORE  LB FF3F|  RESTR1   LB FF44|  RGDSP1   LB FADA|  RMNEM    AB 002D|  RNDH     AB 004F|
RNDL     AB 004E|  RTBL     LB FB19|  RTMASK   LB F80C|  RTMSKZ   LB F87F|  RTS1     LB F831|
RTS2     LB F961|  RTS2B    LB FBEF|  RTS3     LB FBFC|  RTS4     LB FC2B|  RTS4B    LB FCC8|
RTS4C    LB FDC5|  RTS5     LB FE17|  SAV1     LB FF4C|  SAVE     LB FF4A|  SCRL1    LB FC76|
SCRL2    LB FC8C|  SCRL3    LB FC95|  SCRN     LB F871|  SCRN2    LB F879|  SCROLL   LB FC70|
SETAN0   AB C058|  SETAN1   AB C05A|  SETAN2   AB C05C|  SETAN3   AB C05E|  SETCOL   LB F864|
SETGR    LB FB40|  SETIFLG  LB FE86|  SETINV   LB FE80|  SETKBD   LB FE89|  SETMDZ   LB FE1D|
SETMODE  LB FE18|  SETNORM  LB FE84|  SETNXTCO LB F85F|  SETPG3   LB FAA9|  SETPLP   LB FAA8|
SETPWRC  LB FB6F|  SETTXT   LB FB39|  SETVID   LB FE93|  SETWND   LB FB4B|  SLOOP    LB FABA|
SOFTEV   AB 03F2|  SPKR     AB C030|  SPNT     AB 0049|  STACKTOP AB 01FF|  STATUS   AB 0048|
STEPZ    LB FEC4|  STITLE   LB FB65|  STOR     LB FE0B|  STORADV  LB FBF0|  SUBTBL   LB FFE3|
TABV     LB FB5B|  TITLE    LB FB09|  TITLELEN AB 0008|  TOSUB    LB FFBE|  TRACE    LB FEC2|
TXTCLR   AB C050|  TXTSET   AB C051|  UP       LB FC1A|  USR      LB FECA|  USRADR   AB 03F8|
V2       AB 002D|  VERIFY   LB FE36|  VFYOK    LB FE58|  VIDOUT   LB FBFD|  VIDWAIT  LB FB78|
VLINE    LB F828|  VLINEZ   LB F826|  VTAB     LB FC22|  VTABZ    LB FC24|  WAIT     LB FCA8|
WAIT2    LB FCA9|  WAIT3    LB FCAA|  WNDBOTTO AB 0023|  WNDLEFT  AB 0020|  WNDTOP   AB 0022|
WNDWIDTH AB 0021|  WRITE    LB FECD|  XAM      LB FD83|  XAMB     LB FDA3|  XAMPM    LB FDC6|
XBASIC   LB FEB0|  XLTBL    LB FB11|  XREG     AB 0046|  YREG     AB 0047|  YSAV     AB 0034|
YSAV1    AB 0035|  ZMODE    LB FFC7|

                       Apple /// Apple ][ Emulation ROM Source : Formatted Listing
```

```
Assembly complete:      2181 lines
0   Errors flagged on this Assembly

6502 OPCODE STATIC FREQUENCIES

    ADC :   16 | *******
    AND :   16 | *******
    ASL :   18 | ********
    BCC :   30 | **************
    BCS :   10 | ****
    BEQ :   27 | ************
    BIT :    5 | **
    BMI :    6 | **
    BNE :   43 | ********************
    BPL :   14 | ******
    BRK :   34 | ****************
    CLC :    1 |
    CLD :    4 | *
    CMP :   35 | ****************
    CPX :    3 | *
    CPY :    6 | **
    DEC :    6 | **
    DEX :   13 | ******
    DEY :   13 | ******
    EOR :    8 | ***
    INC :   10 | ****
    INX :    6 | **
    INY :    7 | ***
    JMP :   21 | *********
    JSR :   92 | *******************************************
    LDA :  127 | ************************************************************
    LDX :   16 | *******
    LDY :   37 | *****************
    LSR :   20 | *********
    NOP :   76 | ************************************
    ORA :   11 | *****
    PHA :   20 | *********
    PHP :    3 | *
    PLA :   21 | *********
    PLP :    4 | *
    ROL :    6 | **
    ROR :    1 |
    RTI :    1 |
    RTS :   38 | ******************
    SBC :    4 | *
    SEC :    4 | *
    STA :   63 | *****************************
    STX :    5 | **
    STY :   10 | ****
    TAX :    7 | ***
    TAY :    5 | **
    TSX :    1 |
    TXA :    4 | *
    TYA :    3 | *
```

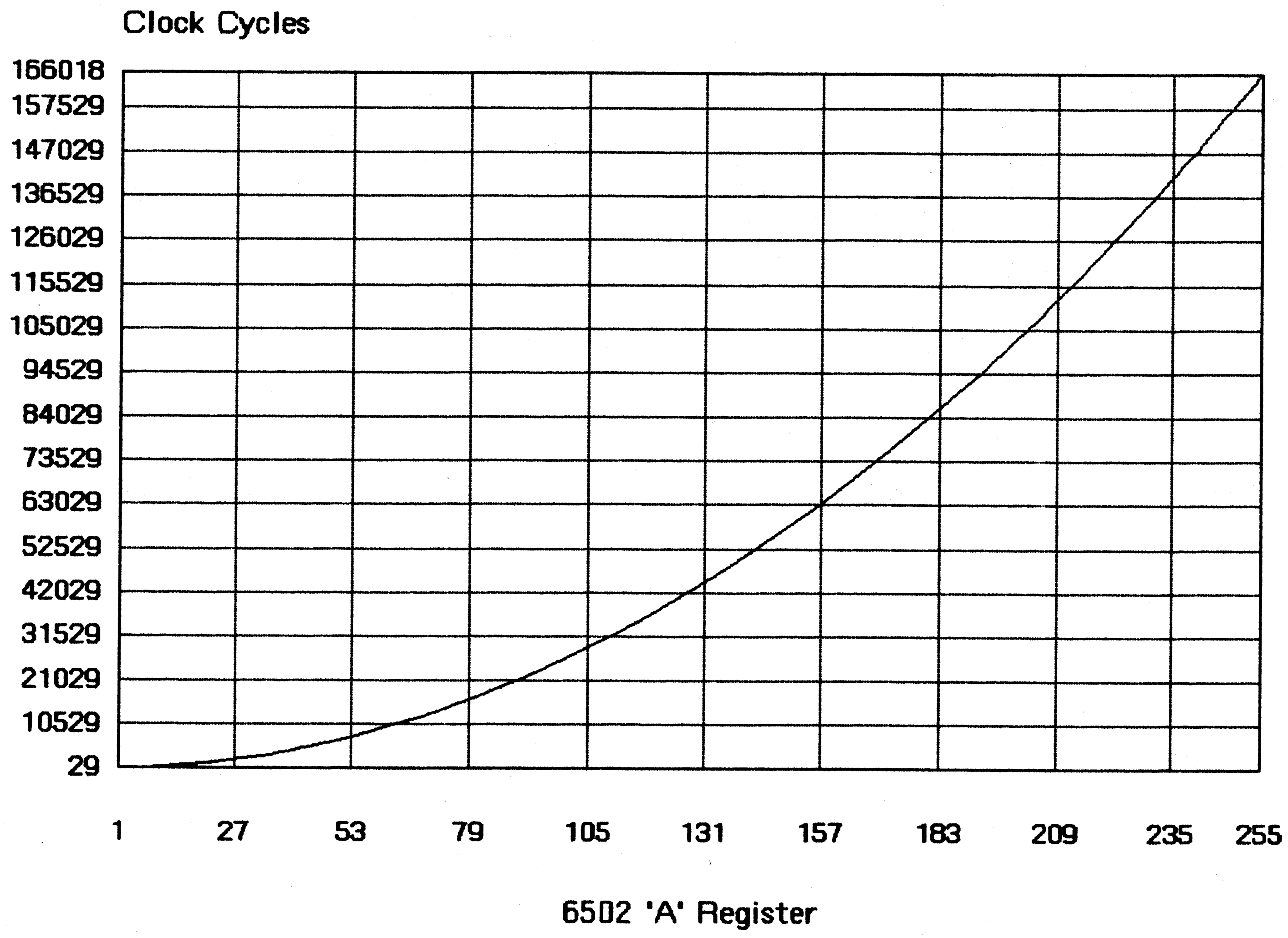


```
   Unused opcodes:

   BVC  BVS  CLI  CLV  SED  SEI  TXS

   Program opcode usage:  87 %

(0.99) That's all, Folks ...
```

Apple /// Apple][Emulation ROM Source : Formatted Listing <<< F I N I S >>>

Apple][Monitor ROM WAIT Routine

A Register vs. Clock Cycles



Clock Cycles

[Prepared by David Craig (736 Edgewater, Wichita, Kansas 67230)]

"_48.PICT" 557 KB 2001-08-13 dpi: 600h x 600v pix: 4261h x 5574v