

```

C100:      FEC5  3  FUNCXIT EQU F8ORG+$6C5 ;RETURN ADDRESS
C100:      PCF0  4  MINI  EQU  F8ORG+$4F0
C100:      5  *
C100:      6  * BASIC FUNCTION HOOK:
C100:      7  *
C100:      8  * $C100 is called by the patched $F8 ROM.
C100:      9  * It provides an extension to $F8 routines
C100:     10  * that do not work in 80 columns.
C100:     11  *
C100:     12  * Before jumping here, the $F8 rom disabled
C100:     13  * slot I/O and enabled ROM I/O. This makes
C100:     14  * the entire space from $C100 - $CFFF with the
C100:     15  * exception of the $C300 page available.
C100:     16  *
C100:     17  * On exit slot I/O is restored if necessary.
C100:     18  *
C100:     19  * INPUT: Y=FUNCTION AS FOLLOWS:
C100:     20  *
C100:     21  *      1 = KEYIN
C100:     22  *      2 = Fix escape char
C100:     23  *      3 = BASCALC
C100:     24  *      4 = VTAB or VTABZ
C100:     25  *      5 = HOME
C100:     26  *      6 = SCROLL
C100:     27  *      7 = CLREOL
C100:     28  *      8 = CLREOLZ
C100:     29  *      9 = RESET
C100:     30  *      A = CLREOP
C100:     31  *      B = RDKEY
C100:     32  *      C = SETWND
C100:     33  *      D = Mini Assembler
C100:     34  *      E = set 40 columns on PR#0/IN#0
C100:     35  *      F = Fix pick for monitor
C100:     36  *
C100:     37  * Stack has PHP for status of internal $CND0 ROM
C100:     38  *
C100:     39  * Note: If 80 Vid is on and the MODE byte is valid,
C100:     40  * this call will be dispatched to an 80 column routine
C100:     41  * by B.FUNCO. Otherwise it will be dispatched to a
C100:     42  * 40 column routine by B.OLDFUNC. In all cases return
C100:     43  * to the Autostart ROM is done through F.RETURN.
C100:     44  *
C100:4C 13 C2  45  B.FUNC JMP  DISPATCH ;figure out what to do
C103:      46  *
C103:A4 24  47  F.CLREOP LDY CH      ; ESC F IS CLR TO END OF PAGE
C105:A5 25  48          LDA  CV
C107:48      49  CLEOPI PHA
C108:20 03 CE 50          JSR  VTABZ
C10B:20 F4 C1 51          JSR  X.CLREOLZ
C10E:A0 00      52          LDY  #$00
C110:68      53          PLA
C111:69 00      54          ADC  #$00      ;(carry set)
C113:C5 23      55          CMP  WNDBTM
C115:90 F0 C107 56          BCC  CLEOPI

```

```

C117:B0 34 C14D 57          BCS  GVTZ      ;=>always to VTABZ
C119:      58  *
C119:A5 22      59  F.HOME LDA  WNDTOP
C11B:85 25      60          STA  CV
C11D:A0 00      61          LDY  #$00
C11F:84 24      62          STY  CH
C121:F0 E4 C107 63          BEQ  CLEOPI      ;(ALWAYS TAKEN)
C123:      64  *
C123:A5 22      65  F.SCROLL LDA  WNDTOP
C125:48      66          PHA
C126:20 03 CE  67          JSR  VTABZ
C129:A5 28      68  SCRL1 LDA  BASL
C12B:85 2A      69          STA  BAS2L
C12D:A5 29      70          LDA  BASH
C12F:85 2B      71          STA  BAS2H
C131:A4 21      72          LDY  WNDWDTH
C133:88      73          DEY
C134:68      74          PLA
C135:69 01      75          ADC  #$01
C137:C5 23      76          CMP  WNDBTM
C139:B0 0D C148 77          BCS  SCRL3
C13B:48      78          PHA
C13C:20 03 CE  79          JSR  VTABZ
C13F:B1 28      80  SCRL2 LDA  (BASL),Y
C141:91 2A      81          STA  (BAS2L),Y
C143:88      82          DEY
C144:10 F9 C13F 83          BPL  SCRL2
C146:30 E1 C129 84          BMI  SCRL1
C148:A0 00      85  SCRL3 LDY  #$00
C14A:20 F4 C1  86          JSR  X.CLREOLZ
C14D:A5 25      87  GVTZ  LDA  CV
C14F:4C 03 CE  88  GVTZ2 JMP  VTABZ      ;set vertical base
C152:      89  *
C152:      C152 90  F.SETWND EQU  *
C152:A9 28      91          LDA  #40
C154:85 21      92          STA  WNDWDTH
C156:A9 18      93          LDA  #24
C158:85 23      94          STA  WNDBTM
C15A:A9 17      95          LDA  #23
C15C:85 25      96          STA  CV
C15E:D0 EF C14F 97          BNE  GVTZ2      ;=>go do vtab, exit
C160:      98  *
C160:      99  * Load Y from BAS2L and clear line
C160:     100  *
C160:A4 2A     101  F.CLREOLZ LDY BAS2L      ;set up by $F8 ROM
C162:4C F4 C1  102          JMP  X.CLREOLZ ;and clear line
C165:     103  *
C165:     104  * 80 column routines begin here
C165:     105  *
C165:4C EB CB  106  B.SCROLL JMP SCROLLUP ;DO IT FOR CALLER
C168:     107  *
C168:     108  * Clear to end of line using Y = OURCH
C168:     109  *
C168:4C 9A CC  110  B.CLREOL JMP X.GS      ;clear to end of line

```