

```

C168:      111 *
C168:      112 * Clear to end of line using Y = BASZL
C168:      113 * which was set up by the $F8 ROM
C168:      114 *
C168:A4 2A 115 B.CLRZLDZ LDY BASZL ;get Y
C16D:4C 9D CC 116 JMP X.GSEOLZ ;clear to end of line
C170:      117 *
C170:4C 74 CC 118 B.CLRZLDZ JMP X.VT ;CLEAR TO EOS
C173:4C A0 C2 119 B.SETWND JMP B.SETWNDX
C176:4C B0 C2 120 B.RESET JMP B.RESETX ;MUST BE IN BFUNC PAGE
C179:4C F2 C2 121 B.RDKEY JMP B.RDKEYX
C17C:      122 *
C17C:20 90 CC 123 B.HOME JSR X.FF ;HOME & CLEAR
C17F:AD 7B 05 124 LDA OURCH
C182:85 24 125 STA CH ;COPY CH/CV FOR CALLER
C184:8D 7B 04 126 STA OLDCH ;REMEMBER WHAT WE SET
C187:4C FE CD 127 JMP VTAB ;calc base & return
C18A:      128 *
C18A:      129 * Complete PR# or IN# call. Quit video firmware
C18A:      130 * if PR#0 and it was active (B.QUIT). Complete call
C18A:      131 * if inactive (F.QUIT).
C18A:      132 *
C18A:      C18A 133 B.QUIT EQU *
C18A:B4 00 134 LDY LOCO,X ;was it PR#0/IN#0?
C18C:F0 0F C19D 135 BEQ NOTO ;=>no, not slot 0
C18E:C0 1B 136 CPY #KEYIN ;was it IN#0?
C190:F0 0E C1A0 137 BEQ ISO ;=>yes, update high byte
C192:20 80 CD 138 JSR QUIT ;quit the firmware
C195:B4 00 139 F.QUIT LDY LOCO,X ;get low byte into Y
C197:F0 04 C19D 140 BEQ NOTO ;not slot 0, firmware inactive
C199:A9 FD 141 F8HOOK LDA #<KEYIN ;set high byte to $FD
C19B:95 01 142 STA LOCL,X
C19D:85 01 143 NOTO LDA LOCL,X ;restore accumulator
C19F:60 144 RTS
C1A0:      145 *
C1A0:A5 37 146 ISO LDA CSHW ;is $C3 in output hook?
C1A2:C9 C3 147 CMP #<BASICIN
C1A4:D0 F3 C199 148 BNE F8HOOK ;=>no, set to $FDOC
C1A6:4C 32 C8 149 JMP C3IN ;else set to $C305, exit A=$C3
C1A9:      150 *
C1A9:A4 24 151 F.RDKEY LDY CH ;else do normal 40 cursor
C1AB:81 28 152 LDA (BASL),Y ;grab the character
C1AD:48 153 PHA
C1AE:29 3F 154 AND #$3F ;set screen to flash
C1B0:09 40 155 ORA #$40
C1B2:91 28 156 STA (BASL),Y ;and display it
C1B4:68 157 F.NOCUR PLA
C1B5:60 158 RTS ;return (A=char)
C1B6:      159 *
C1B6:A8 160 F.BASCALC TAY ;restore Y
C1B7:A5 28 161 LDA BASL ;restore A
C1B9:20 BA CA 162 JSR BASCALC ;calculate base address
C1BC:90 4C C20A 163 BCC F.RETURN ;BASCALC always returns BCC!
C1BE:      164 *

```

```

C1BE:      C1BE 165 B.ESCFIX EQU *
C1BE:20 14 CE 166 JSR UPSHFT ;upshift lowercase
C1C1:A0 03 167 B.ESCFIX1 LDY #4-1 ;SCAN FOR A MATCH
C1C3:      C1C3 168 B.ESCFIX2 EQU *
C1C3:D9 EE C2 169 CMP ESCIN,Y ;IS IT?
C1C5:D0 03 C1CB 170 BNE B.ESCFIX3 ;=>NAW
C1C8:B9 A4 C9 171 LDA ESCOUT,Y ;YES, TRANSLATE IT
C1CB:      C1CB 172 B.ESCFIX3 EQU *
C1CB:88 173 DEY
C1CC:10 F5 C1C3 174 BPL B.ESCFIX2
C1CE:30 3A C20A 175 BMI F.RETURN ;RETURN:CHAR IN AC
C1D0:      176 *
C1D0:20 70 C8 177 F.BOUT JSR ROUT ;print the character
C1D3:4C 0A C2 178 JMP F.RETURN ;AND RETURN
C1D6:      179 *
C1D6:      180 * Do displaced mnemonic stuff
C1D6:      181 *
C1D6:8A 182 MNNDX TXA ;get old acc
C1D7:29 03 183 AND #$03 ;make it a length
C1D9:85 2F 184 STA LENGTH
C1DB:A5 2A 185 LDA BASZL ;get old Y into A
C1DD:29 8F 186 AND #$8F
C1DF:4C 71 CA 187 JMP DOMN ;and go to open spaces
C1E2:      188 *
C1E2:20 F0 FC 189 GOMINI JSR MINI ;do mini-assembler
C1E5:8A 190 TXA ;X=0. Set mode to 0, and counter
C1E6:85 34 191 STA YSAV ;so not CR on new line
C1E8:60 192 RTS
C1E9:      193 *
C1E9:      194 * Pick an 80 column character for the monitor
C1E9:      195 *
C1E9:AC 7B 05 196 FIXPICK LDY OURCH ;get 80 column cursor
C1EC:20 44 CE 197 JSR PICK ;pick the character
C1EF:09 80 198 ORA #$80 ;always pick as normal
C1F1:60 199 RTS ;and return
C1F2:      200 *
C1F2:      201 * Load CH into Y and clear line
C1F2:      202 *
C1F2:      C1F2 203 F.CLRZLDZ EQU *
C1F2:A4 24 204 X.CLRZLDZ LDY CH ;get horizontal position
C1F4:A9 A0 205 X.CLRZLDZ LDA #$A0 ;store a normal blank
C1F6:2C 1E C0 206 BIT ALTCHARSET ;unless alternate char set
C1F9:10 06 C201 207 BPL X.CLRZLDZ
C1FB:24 32 208 BIT INVFLG ;and inverse
C1FD:30 02 C201 209 BMI X.CLRZLDZ
C1FF:A9 20 210 LDA #$20 ;use inverse blank
C201:4C A8 CC 211 X.CLRZLDZ JMP CLR40 ;clear to end of line
C204:      212 *
C204:      213 * Call VTAB or VTABZ for 40 or 80 columns. Acc (CV)
C204:      214 * is saved in BASL.
C204:      215 *
C204:A8 216 F.VTABZ TAY ;restore Y
C205:A5 28 217 LDA BASL ;and A
C207:20 03 CE 218 JSR VTABZ ;do VTABZ

```