

```

FD16:EA      301      NOP
FD17:EA      302      NOP
FD18:6C 38 00 303 RDKEY1 JMP (KSWL) ;GO TO USER KEY-IN
FD18:        304 *
FD18:        FD1B 305 KEYIN EQU *
FD18:A0 03     306 LDY #3 ;RDKEY/RAA0981
FD1D:4C B4 FB 307 GOTOCX2 JMP GOTOCX ;/RAA0981
FD20:EA      308      NOP ;/RAA0981
FD21:        309 *
FD21:        FD21 310 RDESC EQU *
FD21:20 0C FD 311 JSR RDKEY ;GET A KEY
FD24:A0 01     312 LDY #1 ;CODE=FIXIT
FD26:D0 F5 FD1D 313 BNE GOTOCX2 ;=>always
FD28:        314 *
FD28:        315 * Flag to the video firmware that escapes are allowed.
FD28:        316 * This routine is called by RDCHAR which is called by
FD28:        317 * GETLN. The high bit of MSLOT is set by all cards
FD28:        318 * that use the C800 space.
FD28:        319 *
FD28:4E F8 07 320 NEWRDKEY LSR MSLOT ;<128 means escape allowed
FD28:4C 0C FD 321 JMP RDKEY ;now read the key
FD2E:EA      322      NOP
FD2F:        323 *
FD2F:20 21 FD 324 ESC JSR RDESC ;/RAA0981
FD32:20 A5 FB 325 JSR ESCNEW ;HANDLE ESC FUNCTION.
FD35:20 28 FD 326 RDCHAR JSR NEWRDKEY ;Flag RDCHAR and read key
FD38:C9 9B     327 CMP #9B ;'ESC'?
FD3A:F0 F3 FD2F 328 BEQ ESC ; YES, DON'T RETURN.
FD3C:60      329 RTS
FD3D:        330 *
FD3D:A0 0F     331 PICKFIX LDY #9F ;code = fixpick
FD3F:20 B4 FB 332 JSR GOTOCX ;do 80 column pick
FD42:A4 24     333 LDY CH ;restore Y
FD44:9D 00 02 334 STA IN,X ;and save new character
FD47:        335 *#03 AUTOST2 Auto-Start Monitor ROM 27-AUG-84

```

PAGE 20

```

FD6A:A5 33     354 GETLN LDA PROMPT ;OUTPUT PROMPT CHAR
FD6C:20 ED FD 355 JSR COUT
FD6F:A2 01     356 LDY #01 ;INIT INPUT INDEX
FD71:8A      357 BCKSPC TKA
FD72:F0 F3 FD67 358 BEQ GETLNZ ;WILL BACKSPACE TO 0
FD74:CA      359 DEX
FD75:20 35 FD 360 NXTCHAR JSR RDCHAR
FD78:C9 95     361 CMP #95 ;USE SCREEN CHAR
FD7A:D0 08 FD84 362 BNE ADDINP ; FOR CONTROL-U
FD7C:B1 28     363 LDA (BASL),Y ;do 40 column pick
FD7E:2C 1F C0 364 BIT RD80VID ;80 columns?
FD81:30 BA FD3D 365 BMI PICKFIX ;=>yes, fix it
FD83:EA      366 NOP
FD84:9D 00 02 367 ADDINP STA IN,X ;ADD TO INPUT BUFFER
FD87:C9 8D     368 CMP #8D
FD89:D0 BC FD47 369 BNE NOTCR
FD8B:20 9C FC 370 JSR CLREOL ;CLR TO EOL IF CR
FD8E:A9 8D     371 CROUT LDA #8D
FD90:D0 5B FD8D 372 BNE COUT ;(ALWAYS)
FD92:        373 *
FD92:A4 3D     374 PRAI LDY AIH ;PRINT CR,AI IN HEX
FD94:A6 3C     375 LDX AIL
FD96:20 8E FD 376 PRYX2 JSR CROUT
FD99:20 40 F9 377 LDX PRNTYX
FD9C:A0 00     378 LDY #00
FD9E:A9 AD     379 LDA #8A ;PRINT '-'
FDA0:4C ED FD 380 JMP COUT
FDA3:        381 *
FDA3:A5 3C     382 XAMB LDA AIL
FDA5:09 07     383 ORA #07 ;SET TO FINISH AT
FDA7:85 3E     384 STA A2L ; MOD 8=7
FDA9:A5 3D     385 LDA AIH
FDAB:85 3F     386 STA A2H
FDAD:A5 3C     387 MO
D8CHK LDA AIL
FDAF:29 07     388 AND #07
FDB1:D0 03 FDB6 389 BNE DATAOUT
FDB3:20 92 FD 390 XAM LDA PRAI
FDB6:A9 A0     391 DATAOUT LDA #8A
FDB8:20 ED FD 392 JSR COUT ;OUTPUT BLANK
FDBB:B1 3C     393 LDA (AIL),Y
FDBD:20 DA FD 394 JSR PRBYTE ;OUTPUT BYTE IN HEX
FDC0:20 BA FC 395 JSR NXTAL
FDC3:90 E8 FDAD 396 BEQ MOD8CHK ;NOT DONE YET. GO CHECK MOD 8
FDC5:60      397 RTS4C RTS ;DONE.
FDC6:        398 *
FDC6:4A      399 XAMPM LSR A
FDC7:90 EA FDB3 400 BEQ XAM ; DETERMINE IF MONITOR MODE IS
FDC9:4A      401 LSR A ; EXAMINE, ADD OR SUBTRACT
FDCA:4A      402 LSR A
FDCB:A5 3E     403 LDA A2L
FDCD:90 02 FDD1 404 BEQ ADD
FDCF:49 FF     405 EOR #5F ;FORM 2'S COMPLEMENT FOR SUBTRACT.
FDD1:65 3C     406 ADD ADC AIL

```