

**** ROMS -- REVISION 2.0 ****

CONTROL KEYS

- CTRL E sets the double spaced EDIT MODE. Hit CTRL E again to EXIT the edit mode.
- CTRL > moves the CURSOR to the right.
- CTRL < moves the CURSOR to the LEFT.
- CTRL U moves the CURSOR UP.
- CTRL D moves the CURSOR DOWN.
- CTRL R DELETES the character or space over which the CURSOR is located. If the R key is kept depressed it will continue to delete.
- CTRL I INSERTS a space at the character or space over which the CURSOR is located and moves that point and all others to the right. You can then enter any characters in the space.
- CTRL N DELETES ALL from under the CURSOR to the end of that line.
- CTRL B BLOCK DELETES whole lines. You hit CTRL B, then #####-#### (FROM#-TO#) then RETURN KEY, where ##### is a **FOUR DIGIT** Line Number. Those lines and ALL BETWEEN will be DELETED!
- CTRL S STOPS SCROLL of LIST as before but only at the end of a line. No more partial lines!
- CTRL Q CONTINUES SCROLL of LIST.
- CTRL V allows screen viewing of the cassette port while depressed (stops viewing on release).
- LINEFEED will ADVANCE the line #'s by 10 each time it is depressed. This enters it to BASIC and prints line # on the screen.
- ESC RESETS LINE COUNTER to 0000. USE CAUTION!!
- RUBOUT gives INSTANT SCREEN CLEAR at any time input from the keyboard can be accepted.

EDITING FUNCTIONS

The now resident EDITOR will operate at any time you are using BASIC-IN-ROM. It will not operate under the DOS(DISK OF SYSTEM) and any attempted access of the DOS will require a POKE49153,0 to allow the EDITOR to be called. You may EDIT any line LISTED in the DOUBLE SPACED FORMAT. You may EDIT the line you are inputting as long as the LINE NUMBER is PRECEDED BY A SPACE!! EDIT does not occur until you hit RETURN.

-/-

CURSOR

The operating system supports two cursors:

- 1.) If you already have our PROGRAMMABLE CHARACTER GENERATOR, it will give you a 'smart' CURSOR so that whatever character is beneath the CURSOR will be centred and show through in REVERSE VIDEO!!
- 2.) If you do not yet have our PROGRAMMABLE CHARACTER GENERATOR, then you will have a FLASHING CURSOR which alternately flashes on and off showing the character beneath it.

Note that the rate of 'flash' (if no video mod) or the reverse video image (with video mod) is set by POKEing a number between 0 and 255 into location DEC 558. Try POKE558,0 for super fast, POKE558,127 for normal, or POKE558,255 for super slow. (Cold Start sets it at 5).

SCREEN WIDTH

The video routines now support only 32 or 64 character screens. If the programmable character generator is in, the EPROM will read the OSI character generator into the programmable character generator and turn it on.

CTRL E

Please note that when the EDITOR is used, BASIC's Terminal width must be in the normal 72 characters. This control flip flops from 0 to 72. (e.g.) If 72 is in the Terminal Width Register (location 15) then CTRL E flips to 0, or vice versa. To EXIT hit CTRL E.

PRINT AT

The operating system will now support PRINT AT.

An example of this would be:

```
0010 POKE 527,75:POKE 528,209:INPUT"TESTING":A$
```

The word TESTING? should appear in the middle of your 32 Chr. Screen Format, asking for A\$ input.

The PRINT AT utility is a feature used to format the screen for GAMES or BUSINESS applications. The two LOCATIONS given contain the LO and HI ADDRESS of the CURSOR POSITION. For C1P Systems with either 32 or 64 Chrs per line, the cursor can be at any location from \$D000 to \$D7FF. For exact locations see OSI manual.

LO ADDRESS OF CURSOR LOCATION=\$020F or DEC. 527

HI ADDRESS OF CURSOR LOCATION=\$0210 or DEC. 528

SELECTABLE SCROLL WINDOW

This feature allows the user to select which portion of the screen will be affected by the scrolling function of the CRT Driver and which can only be changed by direct memory POKES. An example would be the STARFIGHTERS GAME where a real time graphics display fills the upper half of the screen and the STAR SHIPS STATUS can scroll up from the bottom. POKE 547 (or \$0223) with a number from 208 (TOP OF SCREEN) to 211 (BOTTOM OF SCREEN) on Standard C1P, or 208 to 215 if you have the 64 CHR Screen. SELECTABLE SCROLL=\$0223 or DEC 547. Try:

```
0010 POKE547,210 and you will see the scroll disappear part way up the screen. POKE547,208 resets it.
```

FAST SCREEN CLEAR

Try RUBOUT KEY or in program- 0010 POKE CHR\$(127)
For machine language it's JSR \$FCDE.

Installation Instructions for the new PC ROMS

(for Superboard II please turn over)

1.) Monitor ROM

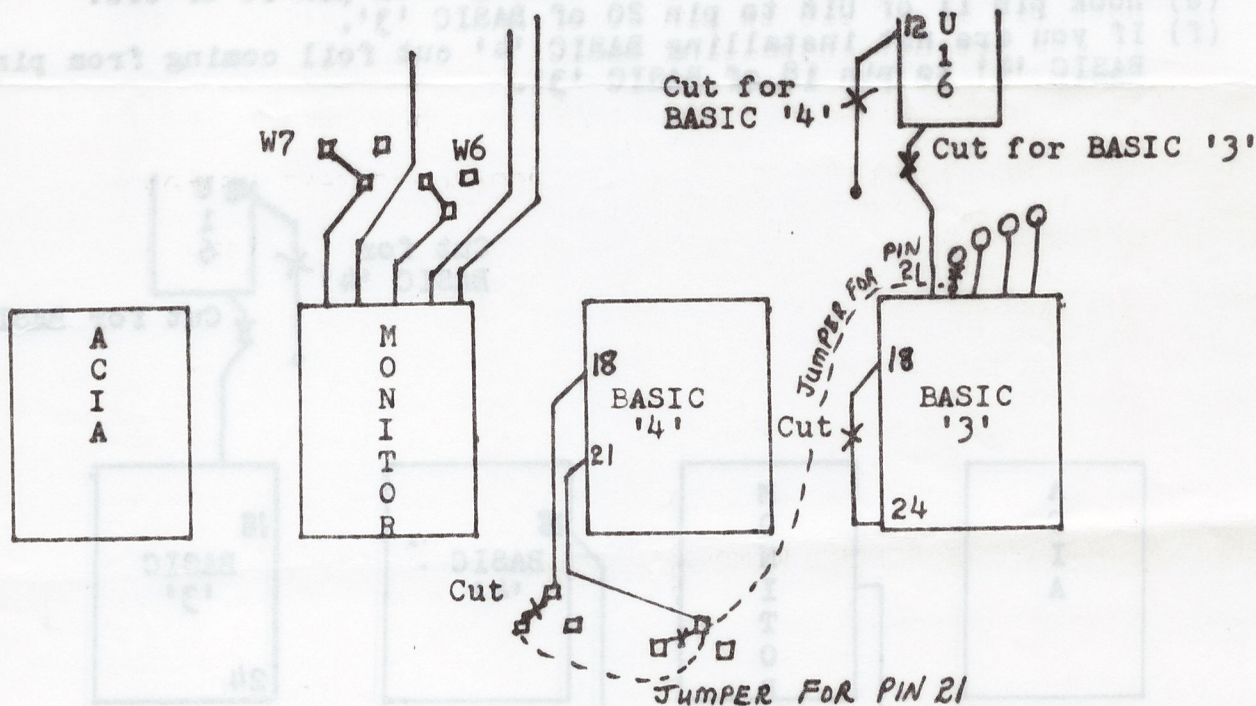
- Cut foil between pins 20 & 21.
Hook a jumper from pin 21 to +5 volts.
- Now look on the diagram below.
- Make sure W7 & W6 are hooked as shown.

2.) BASIC '4'

- Cut foil coming from pin 18 of BASIC 4 as shown.
- Hook pin 18 to pin 20 of BASIC '4'
- Hook pin 13 of U16 to pin 20.
- Cut foil coming from pin 12 of U16 as shown.

3.) BASIC '3'

- Cut foil at pin 18 going towards BASIC '2' on foil side.
- Hook pin 18 of BASIC '2' to +5 volts.
- Cut foil coming from pin 18 of BASIC '3' on component side.
- Hook pin 18 to pin 20 of BASIC '3'.
- Cut foil as shown in diagram coming from pin 10 of U16.
- Hook pin 11 of U16 to pin 20 of BASIC '3'.



KEYBOARD

NOTE: Whether you install one or all of the ROMS you should trace out the foil pattern of pin 21 of the BASIC chips, disconnect them from #2 and hook them to +5V. Make sure that the foil path for #2 stays intact! This change will greatly improve the operation of your ROMS, especially at 2 MHz operation. Look at the schematics above and make the two jumper and foil cuts for pin 21 of the BASIC chips, one on BASIC 4 and the other at the back of BASIC 3.

Installation Instructions for the new PC ROMS

MOD II

(for the model II superboard)

1.) Monitor ROM (On component side)

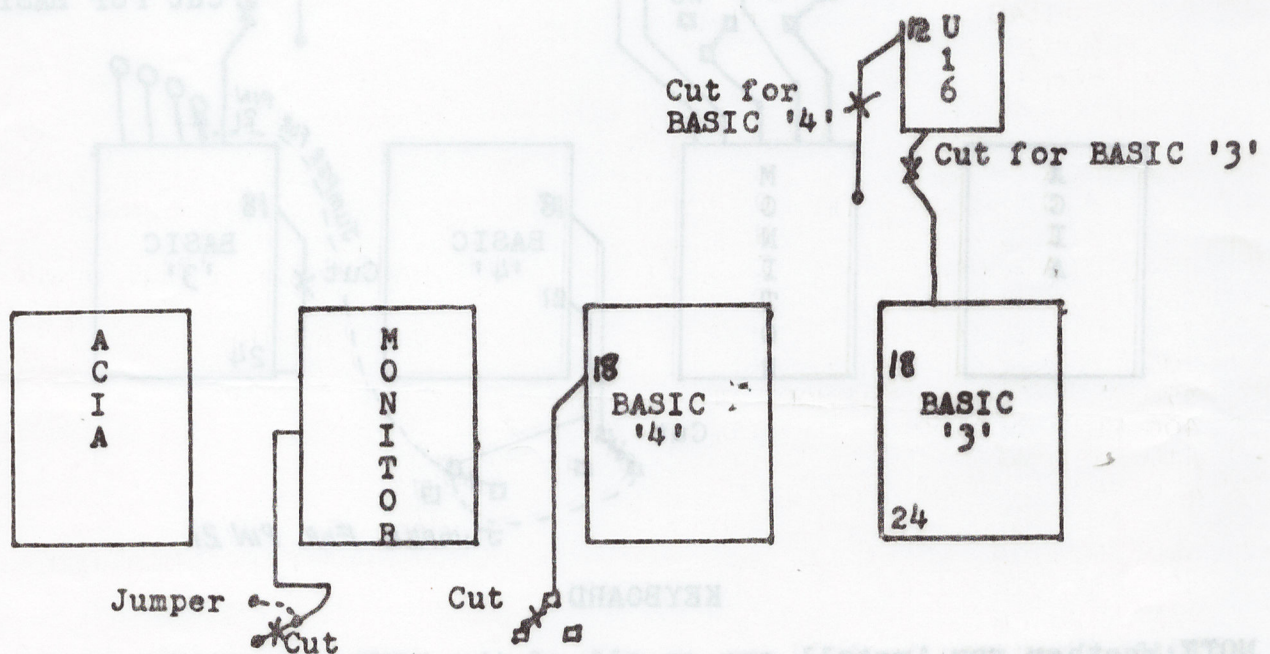
- (a) Cut Jumper coming from pin 21 to ground.
- (b) Hook a jumper from pin 21 to +5 volts.

2.) Basic '4'

- (a) Cut foil coming from pin 18 of BASIC 4 as shown.
- (b) Hook pin 18 to pin 20 of BASIC '4'.
- (c) Hook pin 13 of U16 to pin 20.
- (d) Cut foil coming from pin 12 of U16 as shown.
- (e) Cut lead on foil side between pin 18 BASIC '4' and pin 18 BASIC '3' if not installing BASIC '3' then jumper pin 18 BASIC '3' to +5 volts.

3.) BASIC '3'

- (a) Cut foil at pin 18 going towards BASIC '2' on foil side.
- (b) Hook pin 18 of BASIC '2' to +5 volts.
- (c) Hook pin 18 to pin 20 of BASIC '3'.
- (d) Cut foil as shown in diagram coming from pin 10 of U16.
- (e) Hook pin 11 of U16 to pin 20 of BASIC '3'.
- (f) If you are not installing BASIC '4' cut foil coming from pin 18 BASIC '4' to pin 18 of BASIC '3'.



KEYBOARD

.A

```

10      ; INVERSE VIDEO OR FLASHING CURSOR IF THERE ISN'T A
20      ; PROG. CHAR. GENERATOR BOARD INSTALLED
30      ; COPYRIGHT PROGRESSIVE COMPUTING ALL RIGHTS RESERVED
40      ; BY JOSEPH F. ENDRE
50      ;
60      ; CHANGE IN KEYBOARD ROUTINE
70 F000= OFF=$F000
80 FD70 *=OFF+$D70
90 FD70 4C00F8 JMP INVR5
100 FD73 50      .BYTE $50,$43
100 FD74 43
110
120 C001= FLPSTA=$C001
130 0201= SAVER=$201
140 0200= SIZE=$200
150 0224= EDFLG=$224
160 0226= DEL=$226
170 020E= PRINT=$20E
180 020C= BUFF=$20C
190 022E= TIME=$22E
200 0213= TEMP=$213
210 BFCF= UNS=$BFCF
220 BFE8= OK=$BFE8
230
240 F800 *=OFF+$800
250 F800 8D1402 INVR5 STA $214
260 F803 8A      TXA      SAVE REG.
270 F804 48      PHA
280 F805 98      TYA
290 F806 48      PHA
300 F807 AD01C0 LDA FLPSTA  SEE IF DISK WAS USED
310 F80A 3002 BMI CON
320 F80C D051 BNE LOOP  YES DON'T GIVE SPECIAL CURSOR
330 F80E 2066F8 CON JSR DELY IS IT TIME TO UPDATE CURSOR
340 F811 D04C BNE LOOP
350 F813 AD00E8 LDA $E800  DO WE HAVE PROG. CHAR. GENERATOR
360 F816 C95A CMP #$5A
370 F818 D031 BNE FLASH NO GO DO FLASHING CURSOR
380
390 ; SMART CURSOR ***
400 F81A AD0102 LDA SAVER
410 F81D A200 LDX #0
420 F81F 8E0C02 STX BUFF
430 F822 A003 LDY #3
440 F824 0A CALC ASL A  CALCULATE CHARACTER LOCATION
450 F825 2E0C02 ROL BUFF
460 F828 88 DEY
470 F829 D0F9 BNE CALC
480 F82B 8D0B02 STA BUFF-1 SET UP READ AND WRITE ROUTINE
490 F82E A9E8 LDA #$E8
500 F830 6D0C02 ADC BUFF
510 F833 8D0C02 STA BUFF
520 F836 A007 LDY #7
530 F838 A9B9 LDA #$B9
540 F83A 8D0A02 STA BUFF-2

```



```

550 F83D 200A02 CHR1 JSR BUFF-2 READ CHARACTER
560 F840 49FE EOR #$FE INVERT THEN CENTER
570 F842 4A LSR A
580 ;WRITE IT IN CURSOR LOCATION
590 F843 9958ED STA $AB*8+$E800,Y
600 F846 88 DEY
610 F847 10F4 BPL CHR1
620 F849 3014 BMI LOOP
630 ;SET UP READ ROUTINE
640 F84B A9AD FLASH LDA #$AD
650 F84D 8D0E02 STA PRINT
660 F850 200E02 JSR PRINT
670 F853 C9AB CUR CMP #$AB IS IT THE CURSOR
680 F855 D005 BNE CURSW IF NOT THEN WRITE CURSOR
690 F857 20CFBF JSR UNS PULL CHARACTER UNDER CURSOR AND WR
700 F85A D003 BNE LOOP
710 F85C 20EBBF CURSW JSR OK+3
720 F85F 68 LOOP PLA RESTORE REG.
730 F860 A8 TAY
740 F861 68 PLA
750 F862 AA TAX
760 F863 4C04FD RET JMP OFF+$D04
770 F866 EE2502 DELY INC DEL-1
780 F869 D00D BNE XT
790 F86B CE2602 DEC DEL
800 F86E D008 BNE XT
810 F870 AD2E02 LDA TIME
820 F873 8D2602 STA DEL
830 F876 A900 LDA #0
840 F878 60 XT RTS

```



```

10      ;CRT DRIVER FOR BASIC'4'
20      ;SUPPORTS TRUE 32 OR 32/64
30      ;ALSO PARTIAL SCREEN SCROLL
40 FFBA= INPUT=$FFBA
50 FF6C= OUT=$FF6C
60 0207= LDR=$207
70 0200  *=$200
80 0200= SIZE=*
90 020E= PRINT=$20E
100 020F= STORE=PRINT+1
110 020F= POINT=STORE
120 0201= SAVER=**+1
130 0202= DATCHR=**+2
140 0222= LEN=$222
150 0223= STRT=$223
160 0224= EDFLG=$224
170 022B= TEMP=$22B
180 022E= TIME=$22E
190 DF00= KYBRD=$DF00
200 BF2D  *=$BF2D
210 BF2D 8D0202 CROUT STA DATCHR  SAVE CHAR.
220 BF30 48      PHA
230 BF31 8A      TXA
240 BF32 48      PHA
250 BF33 98      TYA
260 BF34 48      PHA
270 BF35 AD0202  LDA DATCHR  RELOAD CHAR.
280 BF38 F050    BEQ EXIT    DO NOT PRINT NULLS
290 BF3A AC0002  LDY SIZE
300 BF3D C97F    CMP #$7F    IS IT RUBOUT
310 BF3F D005    BNE NORUB
320 BF41 A95F    LDA #$5F    LOAD A CR.
330 BF43 20DCFC  JSR CLEAR   ERASE SCREEN
340 BF46 C90A    NORUB CMP #$A
350 BF48 F04C    BEQ LF
360 BF4A C90D    CMP #$D    IS IT CR.
370 BF4C D006    BNE L2
380 BF4E 20DABF  JSR RT
390 BF51 4C8ABF  JMP EXIT
400 BF54 8D0102  L2  STA SAVER
410 BF57 C95F    CMP #$5F    IS IT BACK SPACE.
420 BF59 D00D    BNE NOBS
430 BF5B A920    LDA #$20    YES DO A BACK SPACE.
440 BF5D 8D0102  STA SAVER
450 BF60 20CFBF  JSR UNS
460 BF63 CE0F02  DEC POINT
470 BF66 D00B    BNE CNT
480 BF68 20CFBF  NOBS JSR UNS  NO WRITE CHARACTER
490 BF6B EE0F02  INC POINT
500 BF6E D003    BNE CNT
510 BF70 EE1002  INC PRINT+2
520 BF73 B9FEFC  CNT  LDA SI2,Y
530 BF76 CD1002  CMP PRINT+2
540 BF79 F002    BEQ CRLF
550 BF7B D00A    BNE EXITM
560 BF7D AD2202  CRLF LDA LEN    ON HOME LINE CHECK FOR AUTO-CR.
570 BF80 6980    ADC #$80
580 BF82 CD0F02  CMP POINT

```


590	BF85	F00C		BEQ	AUTCR	
600	BF87	20E0BF	EXITM	JSR	SAV	SAVE NEXT CHAR. ON SCREEN
610	BF8A	68	EXIT	PLA		RESTORE REG.
620	BF8B	A8		TAY		
630	BF8C	68		PLA		
640	BF8D	AA		TAX		
650	BF8E	68		PLA		
660	BF8F	206CFF		JSR	OUT	OUTPUT TO PROPER PORT
670	BF92	60	EXT	RTS		
680			:			
690	BF93	20DDBF	AUTCR	JSR	RTA	RESET HOME.
700	BF96	20CFBF	LF	JSR	UNS	PRINT CHAR.
710	BF99	A207	SCROL	LDR	#7	SETUP SCROLLING ROUTINE.
720	BF9B	20F6BF	CD	JSR	CD1	
730	BF9E	A920		LDA	##20	
740	BFA0	C000		CPY	#0	ADJUST FOR SCREEN SIZE
750	BFA2	F001		BEQ	BG+1	
760	BFA4	0A	BG	ASL	A	
770	BFA5	8D0802		STA	LDR+1	
780	BFA8	AC2302	RST	LDY	STRT	CHECK FOR START OF SCROLL.
790	BFAB	88		DEY		
800	BFAC	C8	INCY	INY		
810	BFAD	8C0902		STY	LDR+2	
820	BFB0	8C0C02		STY	LDR+5	
830	BFB3	200702	LD	JSR	LDR	SCROLL A LINE FROM TOP DOWN.
840	BFB6	E8		INX		
850	BFB7	F0F3		BEQ	INCY	
860	BFB9	10F8		BPL	LD	
870	BFB8	CC1002		CPY	POINT+1	
880	BFBE	D0F3		BNE	LD	
890	BFC0	A980	LINE	LDA	##80	SCROLL TILL END OF HOME LINE
900	BFC2	6D2202		ADC	LEN	
910	BFC5	8D1702		STA	\$217	
920	BFC8	EC1702		CPX	\$217	
930	BFCB	D0E6		BNE	LD	
940	BFGD	F0B8		BEQ	EXITM	
950	BFCF	AD0102	UNS	LDA	SAVER	PRINT SAVED CHAR.
960	BFD2	A28D	STR	LDR	##8D	SETUP ROUTINE.
970	BFD4	8E0E02		STX	PRINT	
980	BFD7	4C0E02		JMP	PRINT	
990	BFDA	20CFBF	RT	JSR	UNS	
1000	BFDD	20ABF8	RTA	JSR	AURST	RESET HOME POSITION OF CURSOR
1010	BFE0	A9AD	SAV	LDA	##AD	
1020	BFE2	8D0E02		STA	PRINT	
1030	BFE5	200E02		JSR	PRINT	
1040	BFE8	8D0102	OK	STA	SAVER	SAVE CHAR. AND WRITE CURSOR.
1050	BFEB	A9AB		LDA	##AB	
1060	BFED	D0E3		BNE	STR	
1070			:			
1080	BFEF			*= \$BFEF		
1090			:	ROUTINE	FOR SCROLL	
1100	BFEF	BD00D0	CODE	LDA	\$D000, X	
1110	BFF2	9D00D0		STA	\$D000, X	
1120	BFF5	60		RTS		
1130	BFF6	BDEEBF	CD1	LDA	CODE-1, X	
1140	BFF9	9D0602		STA	LDR-1, X	
1150	BFFC	CA		DEX		
1160	BFFD	D0F7		BNE	CD1	
1170	BFFF	60		RTS		


```

1180      ; END OF SCREEN DRIVER
1190      ;
1200 F879      *=$F879
1210 F879 8D2402 SETSC STA EDFLG
1220 F87C A207      LDX #7
1230 F87E 20F6BF      JSR CD1
1240 F881 A960      SETCH LDA #$60
1250 F883 8D1102      STA POINT+2
1260 F886 A0E8      LDY #$E8
1270 F888 8C0902 LOP1  STY LDR+2
1280 F88B 8C0C02      STY LDR+5
1290 F88E 200702 LOP2  JSR LDR
1300 F891 E8      INX
1310 F892 D0FA      BNE LOP2
1320 F894 C8      INY
1330 F895 C0F0      CPY #$F0
1340 F897 D0EF      BNE LOP1
1350 F899 8D01F1      STA $F101  TURN ON CHAR. GENERATOR.
1360 F89C A9D0      LDA #$D0  SET START OF SCROLL
1370 F89E 8D2302      STA STRT
1380 F8A1 A905      LDA #5
1390 F8A3 8D2E02      STA TIME  SET FLASH RATE.
1400 F8A6 8A      TXA
1410 F8A7 A8      TAY
1420 F8A8 4C35FF      JMP $FF35
1430 F8AB AC0002 AURST LDY SIZE RESET HOME POSITION OF CURSOR
1440 F8AE B9FEFC      LDA SIZ,Y
1450 F8B1 8D1002      STA POINT+1
1460 F8B4 A980      LDA #$80
1470 F8B6 8D0F02      STA POINT
1480 F8B9 4CBCF8      JMP PATCH
1490      ;
1500 FCD5      *=$419
1510 FCD5 20BAFF START JSR INPUT  INPUT A CHARACTER
1520 FCD8 C97F      CMP #$7F  SHOULD WE ERASE SCREEN?
1530 FCDA D021      BNE DONE
1540 FCDC 48      CLEAR PHA
1550 FCDD A920      LDA #$20
1560 FCDF A200      LDX #0
1570 FCE1 9D00D0 NEX  STA $D000,X
1580 FCE4 9D00D1      STA $D100,X
1590 FCE7 9D00D2      STA $D200,X
1600 FCEA 9D00D3      STA $D300,X
1610 FCED 9D00D4      STA $D400,X
1620 FCF0 9D00D5      STA $D500,X
1630 FCF3 9D00D6      STA $D600,X
1640 FCF6 9D00D7      STA $D700,X
1650 FCF9 E8      INX
1660 FCFA D0E5      BNE NEX
1670 FCFC 68      PLA
1680 FCFD 60      DONE RTS
1690 FCFE D3      SIZ .BYTE $D3,$D7
1690 FCFF D7

```



```

1700      ;
1710      ;
1720 FF1B      **=**+$21B
1730 FF1B 20DCFC      JSR CLEAR
1740 FF1E A93F      LDA #$3F
1750 FF20 2CFFD7      BIT $D7FF DO WE HAVE 64 CHARACTERS ?
1760 FF23 3003      BMI C1
1770 FF25 C8      C2      INY
1780 FF26 D001      BNE PUT
1790 FF28 4A      C1      LSR A
1800 FF29 8D2202      PUT      STA LEN
1810 FF2C 8C0002      STY SIZE
1820 FF2F 20ABF8      JSR AURST
1830 FF32 4C79F8      JMP SETSC
1840 F8BC      **=$F8BC
1850 F8BC 20DCFC      PATCH      JSR CTRL
1860 F8BF 7013      BVS RT1
1870 F8C1 A9F7      SS      LDA #$F7
1880 F8C3 20D5F8      JSR STABIT
1890 F8C6 300C      BMI RT1
1900 F8C8 20DCFC      Q      JSR CTRL
1910 F8CB 70FB      BVS Q
1920 F8CD A9FD      LDA #$FD
1930 F8CF 20D5F8      JSR STABIT
1940 F8D2 30F4      BMI Q
1950 F8D4 60      RT1      RTS
1960 F8D5 8D00DF      STABIT      STA $DFOO
1970 F8D8 2C00DF      BIT $DFOO
1980 F8DB 60      RTS
1990 F8DC A9FE      CTRL      LDA #$FE
2000 F8DE 20D5F8      JSR STABIT
2010 F8E1 60      RTS

```


A

```

10      ;EDITOR ROUTINES FOR PC ROM MONITOR.
20      ;SUPPORTS NEW CRT DRIVER IN BASIC '4' ROM
30      ;COPYRIGHT PROGRESSIVE COMPUTING ALL RIGHTS RESERVED
40 0200      *=$200
50 F000=     OFF=$F000
60 0200=     SIZE=*
70 0201=     SAVER=**+1
80 0202=     DATCHR=**+2
90 0207=     LDR=$207
100 0235=    STR=$235
110 020E=    PRINT=$20E
120 020F=    STORE=PRINT+1
130 020F=    POINT=STORE
140 0213=    TEMP=$213
150 0222=    LEN=$222
160 0223=    STRT=$223
170 0224=    EDFLG=$224
180 0226=    DEL=$226
190 0228=    LINUM=$228
200 022C=    WIDTH=$22C
210 022D=    CONST=$22D
220 022E=    TIME=$22E
230 022F=    COUNT=$22F
240 0230=    INDEX=$230
250 0231=    BLFLAG=$231
260 0232=    TEMPX=$232
270 0233=    TEMP1=$233
280 DF00=    KYBRD=$DF00
290 FCD5=    START=OFF+$CD5
300 FFCB=    ACIAIN=OFF+$FCB
310 BFCE=    UNS=$BFCE
320 BFE0=    SAV=$BFE0
330 FFBA=    INPUT=OFF+$FBA
340 FF6C=    OUT=OFF+$F6C
350 BF2D=    CROUT=$BF2D
360 C001=    FLPSTA=$C001
370 FEF0      *=$OFF+$FE0
380 FEF0 E2    .BYTE $E2,$F8
380 FEF1 F8
390 FEF2 2D    .BYTE $2D,$BF,$9B,$FF
390 FEF3 BF
390 FEF4 9B
390 FEF5 FF
400 FEF6 8B    .BYTE $8B,$FF,$96,$FF
400 FEF7 FF
400 FEF8 96
400 FEF9 FF
410 F8E2      *=$OFF+$8E2
420 F8E2 8A    BEGIN TXA      START OF INPUT ROUTINES
430 F8E3 48      PHA          SAVE REG.
440 F8E4 98      TYA
450 F8E5 48      PHA
460 F8E6 20D5FC INCHR JSR START  INPUT FROM CORRECT PORT.
470 F8E9 8D1302 STA TEMP
480 F8EC AC01C0 LDY FLPSTA WAS DISK USED ?

```



```

490 F8EF 3002          BMI ROMBA
500 F8F1 D01F          BNE NOTED
510 F8F3 2C0302 ROMBA BIT $203    IS IT A LOAD.
520 F8F6 301A          BMI NOTED
530 F8F8 C91B          KEY    CMP ##1B ESC KEY
540 F8FA D021          BNE LF
550 F8FC A930          LDA ##30
560 F8FE A204          LDX #4
570 F900 9D2802 LREST STA LINUM, X
580 F903 CA            DEX
590 F904 10FA          BPL LREST
600 F906 E910          SBC ##10
610 F908 8D2702          STA LINUM-1
620 F90B A0F9          LDY #MSG/256
630 F90D A915          LDA #MSG*256/256
640 F90F 20C3A8          JSR $A8C3
650 F912 4CECF9 NOTED JMP OUT2+3 EXIT
660 F915 41            MSG    .BYTE 'AUTO#', $D, $A, 0
660 F916 55
660 F917 54
660 F918 4F
660 F919 23
660 F91A 0D
660 F91B 0A
660 F91C 00
670 F91D C90A          LF      CMP ##A
680 F91F D051          BNE VW
690 F921 E000          CPX #0 IS IT START OF INPUT BUFFER ?
700 F923 D0ED          BNE NOTED
710 F925 A030          LDY ##30
720 F927 A93A          LDA ##3A
730 F929 D00F          BNE AUTO
740 F92B EE2B02 LINC1 INC LINUM+3    AUTOLINE BY ONE'S
750 F92E A030 LINC0 LDY ##30    AUTOLINE NO INCREMENT
760 F930 A93A          LDA ##3A
770 F932 CD2B02          CMP LINUM+3
780 F935 D01C          BNE ENDL
790 F937 8C2B02          STY LINUM+3
800 F93A EE2A02 AUTO INC LINUM+2    AUTOLINE BY TEN'S
810 F93D CD2A02          CMP LINUM+2
820 F940 D011          BNE ENDL
830 F942 8C2A02          STY LINUM+2
840 F945 EE2902          INC LINUM+1
850 F948 CD2902          CMP LINUM+1
860 F94B D006          BNE ENDL
870 F94D 8C2902          STY LINUM+1
880 F950 EE2802          INC LINUM
890 F953 A200          ENDL   LDX #0    STORE LINE # IN BUFFER AND PRINT IT.
900 F955 BD2702 LOP LDA LINUM-1, X
910 F958 9513          STA $13, X
920 F95A AC3102          LDY BLFLAG
930 F95D F003          BEQ NOPRT
940 F95F 20E5A8          JSR $A8E5
950 F962 E8            NOPRT INX
960 F963 E005          CPX #5
970 F965 D0EE          BNE LOP
980 F967 AD3102          LDA BLFLAG    AUTOLINE OR BLOCK DELETE ?
990 F96A F005          BEQ LEND
1000 F96C 68          PLA

```


1010	F96D	A8		TAY
1020	F96E	68		PLA
1030	F96F	A920		LDA #20
1040	F971	60	LEND	RTS
1050	F972	C916	VW	CMP #16 CNTRL 'V'
1060	F974	D03F		BNE EDIT WAS IT A VIEW COMMAND?
1070	F976	20CBFF	VIEW	JSR ACIAIN SERIAL PORT VIEW.
1080	F979	202DBF		JSR CROUT
1090	F97C	A9FE		LDA #FE
1100	F97E	20D5F8		JSR OFF+\$8D5
1110	F981	50F3		BVC VIEW
1120	F983	4CE6F8		JMP INCHR
1130	F986	20CFBF	ADDR	JSR UNS GENERAL ADD USING CONST.
1140	F989	18		CLC
1150	F98A	AD0F02		LDA POINT
1160	F98D	6D2D02		ADC CONST
1170	F990	8D0F02		STA POINT
1180	F993	9003		BCC RET
1190	F995	EE1002		INC POINT+1
1200	F998	60	RET	RTS
1210	F999	20CFBF	SUBTR	JSR UNS GENERAL SUBTRACT USING CONST
1220	F99C	38		SEC
1230	F99D	AD0F02		LDA POINT
1240	F9A0	ED2D02		SBC CONST
1250	F9A3	8D0F02		STA POINT
1260	F9A6	B003		BCS RET1
1270	F9A8	CE1002		DEC POINT+1
1280	F9AB	60	RET1	RTS
1290	F9AC	20FFF9	ONE	JSR NULL USED TO INC. OR DEC CURSOR
1300	F9AF	A901		LDA #1
1310	F9B1	8D2D02	STOR	STA CONST
1320	F9B4	60		RTS
1330	F9B5	C905	EDIT	CMP #5 CNTRL E TOGGLETERMINAL WIDTH
1340	F9B7	D006		BNE EDT1
1350	F9B9	A50F		LDA \$0F
1360	F9BB	4948		EOR #48
1370	F9BD	850F		STA \$0F
1380	F9BF	AD1302	EDT1	LDA TEMP
1390	F9C2	C9EC		CMP #EC CNTRL <
1400	F9C4	D009		BNE RIGHT
1410	F9C6	20ACF9	LEFT	JSR ONE
1420	F9C9	2099F9		JSR SUBTR
1430	F9CC	4CE6F9		JMP XT
1440	F9CF	C9EE	RIGHT	CMP #EE CNTRL >
1450	F9D1	D009		BNE UP
1460	F9D3	20ACF9	WRT	JSR ONE
1470	F9D6	2086F9		JSR ADDR
1480	F9D9	4CE6F9		JMP XT
1490	F9DC	C915	UP	CMP #15 CNTRL U
1500	F9DE	D028		BNE DOWN
1510	F9E0	20F4F9		JSR MACH
1520	F9E3	2099F9		JSR SUBTR
1530	F9E6	20FFF9	XT	JSR NULL
1540	F9E9	20E0BF	OUT2	JSR SAV SAVE CONTENTS OF SCREEN
1550	F9EC	68		PLA RESTORE REG.
1560	F9ED	A8		TAY
1570	F9EE	68		PLA

1580	F9EF	AA		TAX
1590	F9F0	AD1302		LDA TEMP
1600	F9F3	60		RTS
1610	F9F4	A920	MACH	LDA ##20 SET PROPER INC.FOR YOUR MACH.
1620	F9F6	AC0002		LDY SIZE
1630	F9F9	F001		BEG LRG1+1
1640	F9FB	0A	LRG1	ASL A
1650	F9FC	4CB1F9		JMP STOR
1660	F9FF	A900	NULL	LDA #0
1670	FA01	8D2402		STA EDFLG
1680	FA04	8D1302		STA TEMP
1690	FA07	60		RTS
1700	FA08	C904	DOWN	CMP #4 CNTRL D
1710	FA0A	D009		BNE CR
1720	FA0C	20F4F9		JSR MACH
1730	FA0F	2086F9		JSR ADDR
1740	FA12	4CE6F9		JMP XT
1750	FA15	C90D	CR	CMP ##D
1760	FA17	D019		BNE INSRT
1770	FA19	AC3102		LDY BLFLAG BLOCK DELETE ?
1780	FA1C	D003		BNE EDT
1790	FA1E	4CA2FB		JMP BLDEL
1800	FA21	AC2402	EDT	LDY EDFLG EDIT MODE ?
1810	FA24	D0C6		BNE OUT2+3
1820	FA26	8D2402		STA EDFLG
1830	FA29	8D1302		STA TEMP
1840	FA2C	20F1FA		JSR SETUP
1850	FA2F	4CAFFA		JMP PUT
1860	FA32	C909	INSRT	CMP #9 CNTRL I INSERT MODE?
1870	FA34	D058		BNE DELET
1880	FA36	20F1FA		JSR SETUP FIND START AND END AND LEN
1890	FA39	AA		TAX X= # BYTES TO END OF LINE.
1900	FA3A	A8		TAY
1910	FA3B	208AFB		JSR SET
1920	FA3E	C8		INY
1930	FA3F	200702	LPIN	JSR LDR
1940	FA42	C9AB		CMP ##AB IS IT THE CURSOR ?
1950	FA44	D006		BNE CON
1960	FA46	AD0102		LDA SAVER
1970	FA49	200A02		JSR LDR+3
1980	FA4C	8A	CON	TXA
1990	FA4D	A8		TAY
2000	FA4E	CA		DEX
2010	FA4F	10EE		BPL LPIN
2020	FA51	206AFA		JSR BLNK
2030	FA54	4CECF9	END	JMP OUT2+3
2040	FA57	C90E	BLANK	CMP ##E CNTRL 'N'
2050	FA59	D018		BNE BLOCK
2060	FA5B	20F1FA		JSR SETUP
2070	FA5E	A8		TAY
2080	FA5F	208AFB		JSR SET
2090	FA62	206AFA	LBLNK	JSR BLNK
2100	FA65	88		DEY
2110	FA66	10FA		BPL LBLNK
2120	FA68	30EA		BMI END
2130	FA6A	A920	BLNK	LDA ##20 BLANK A CHARACTER
2140	FA6C	8D0102		STA SAVER

2150	FA6F	200A02		JSR	LDR+3	
2160	FA72	60		RTS		
2170	FA73	C902	BLOCK	CMP	#2 CNTRL 'B'	
2180	FA75	D008		BNE	VALID	
2190	FA77	A900		LDA	#0	
2200	FA79	8D3102		STA	BLFLAG	
2210	FA7C	4CECF9		JMP	OUT2+3	
2220	FA7F	AC2402	VALID	LDY	EDFLG IS IT A VALID INPUT?	
2230	FA82	D0D0		BNE	END	
2240	FA84	C95F		CMP	##5F IS IT A BACK SPACE?	
2250	FA86	F0CC		BEQ	END	
2260	FA88	8D0102		STA	SAVER	
2270	FA8B	4CD3F9		JMP	WRT	
2280	FA8E	C912	DELET	CMP	##12 CNTRL R	
2290	FA90	D0C5		BNE	BLANK	
2300	FA92	20F1FA		JSR	SETUP FIND START END AND LEN.	
2310	FA95	8D2F02		STA	COUNT # OF BYTES TO END OF LINE.	
2320	FA98	A201		LDX	#1	
2330	FA9A	A000		LDY	#0	
2340	FA9C	208AFB		JSR	SET SETUP READ AND WRITE ROUTINE.	
2350	FA9F	200702	LPDL	JSR	LDR	
2360	FAA2	8A		TXA		
2370	FAA3	AB		TAY		
2380	FAA4	EB		INX		
2390	FAA5	EC2F02		CPX	COUNT	
2400	FAA8	30F5		BMI	LPDL	
2410	FAAA	F0F3		BEQ	LPDL	
2420	FAAC	4CE9F9		JMP	OUT2	
2430	FAAF	208AFB	PUT	JSR	SET SETUP READ AND WRITE TO BUFFER.	
2440	FAB2	A09D		LDY	##9D	
2450	FAB4	8C0A02		STY	LDR+3	
2460	FAB7	A200		LDX	#0	
2470	FAB9	8E0C02		STX	LDR+5 SET ZERO PAGE	
2480	FABC	AD3602		LDA	STR+1	
2490	FABF	8D0902		STA	LDR+2	
2500	FAC2	AD3502		LDA	STR	
2510	FAC5	8D0802		STA	LDR+1	
2520	FAC8	A912		LDA	##12	
2530	FACA	8D0B02		STA	LDR+4	
2540	FACD	200702	LOP1	JSR	LDR PUT SCREEN IN BUFFER.	
2550	FAD0	C9AB		CMP	##AB IS IT THE CURSOR?	
2560	FAD2	D006		BNE	NOCUR	
2570	FAD4	AD0102		LDA	SAVER	
2580	FAD7	200A02		JSR	LDR+3 PUT CHAR NOT CURSOR IN BUFF.	
2590	FADA	EB	NOCUR	INX		
2600	FADB	EC3002	CONT	CPX	INDEX	
2610	FADE	30ED		BMI	LOP1	
2620	FAE0	A900		LDA	#0	
2630	FAE2	200702		JSR	LDR	
2640	FAE5	68		PLA		
2650	FAE6	AB		TAY		
2660	FAE7	68		PLA		
2670	FAE8	AE3002		LDX	INDEX	
2680	FAEB	AD1302		LDA	TEMP	
2690	FAEE	60		RTS		
2700	FAEF	E0	CONS	.BYTE	\$E0,\$C0	
2700	FAF0	C0				
2710	FAF1	AD1002	SETUP	LDA	POINT+1 FIND START OF VALID LINE	

2720	FAF4	8D3602		STA STR+1
2730	FAF7	AD0F02		LDA POINT
2740	FAFA	AC0002		LDY SIZE ACCORDING TO YOUR SCREEN SIZE
2750	FAFD	39E00A		AND CONS,Y
2760	FB00	8D3502		STA STR
2770	FB03	A9BD		LDA #BD SETUP READ ROUTINE.
2780	FB05	8D3402		STA STR-1
2790	FB08	A960		LDA #60
2800	FB0A	8D3702		STA STR+2
2810	FB0D	A200	FIND	LDX #0 RESET COUNTER
2820	FB0F	203402		JSR STR-1 READ A CHARACTER
2830	FB12	C920		CMP #20 FIND SPACE BEFORE LINE #
2840	FB14	D00C		BNE DE
2850	FB16	E8		INX
2860	FB17	203402		JSR STR-1
2870	FB1A	C930		CMP #30 MUST BE A NUMBER.
2880	FB1C	3004		BMI DE
2890	FB1E	C93A		CMP #3A
2900	FB20	302D		BMI GO
2910	FB22	20F4F9	DE	JSR MACH SET LINE INC BY SCREEN SIZE.
2920	FB25	AD3502		LDA STR
2930	FB28	38		SEC
2940	FB29	ED2D02		SBC CONST
2950	FB2C	8D3502		STA STR
2960	FB2F	B003		BCS BACK
2970	FB31	CE3602		DEC STR+1
2980	FB34	AD3602	BACK	LDA STR+1
2990	FB37	C9D0		CMP #D0
3000	FB39	B0D2		BCS FIND
3010	FB3B	A0FB	PRNT	LDY #MSSG/256
3020	FB3D	A948		LDA #MSSG*256/256
3030	FB3F	8D3102		STA BLFLAG
3040	FB42	20C3AB		JSR \$A8C3
3050	FB45	4C74A2		JMP \$A274 DO NOT GO BEYOND TOP OF SCREEN.
3060	FB48	0D	MSSG	.BYTE \$D,'ERROR',0
3060	FB49	45		
3060	FB4A	52		
3060	FB4B	52		
3060	FB4C	4F		
3060	FB4D	52		
3060	FB4E	00		
3070	FB4F	205FFB	GO	JSR LNRD
3080	FB52	8E3002		STX INDEX SAVE # OF CHARACTERS.
3090	FB55	8A	TEST	TXA
3100	FB56	18		CLC
3110	FB57	6D3502		ADC STR
3120	FB5A	38		SEC
3130	FB5B	ED0F02		SBC POINT
3140	FB5E	60		RTS RETURN WITH COUNT FROM CURS. TO END
3150	FB5F	A200	LNRD	LDX #0 RESET COUNTER
3160	FB61	E8	LOP2	INX
3170	FB62	E048		CPX #48 DO NOT ALLOW LINES LONGER THAN BUF
3180	FB64	3005		BMI SAFE
3190	FB66	68		PLA
3200	FB67	68		PLA
3210	FB68	4C3BFB		JMP PRNT
3220	FB6B	203402	SAFE	JSR STR-1

3230	FB6E	C920		CMP ##20	FIND SPACE THEN CHECK HOW MANY.
3240	FB70	D0EF		BNE LOP2	
3250	FB72	8A	CHECK	TXA	MUST BE ATLEAST ONE LINE OF SPCS.
3260	FB73	48		FHA	SAVE COUNTS
3270	FB74	A000		LDY #0	
3280	FB76	E8	-LP1	INX	
3290	FB77	C8		INY	
3300	FB78	C020		CPY ##20	
3310	FB7A	F00B		BEQ OK1	
3320	FB7C	203402	NXT	JSR STR-1	
3330	FB7F	C920		CMP ##20	
3340	FB81	F0F3		BEQ LP1	
3350	FB83	68		PLA	
3360	FB84	AA		TAX	
3370	FB85	D0DA		BNE LOP2	NOT A VALID END OF LINE.
3380	FB87	68	OK1	PLA	VALID END RESTORE COUNTS
3390	FB88	AA		TAX	
3400	FB89	60		RTS	
3410	FB8A	AD1002	SET	LDA POINT+1	SETUP READ AND WRITE.
3420	FB8D	8D0902		STA LDR+2	
3430	FB90	8D0C02		STA LDR+5	
3440	FB93	AD0F02		LDA POINT	
3450	FB96	8D0802		STA LDR+1	
3460	FB99	8D0B02		STA LDR+4	
3470	FB9C	A999		LDA ##99	
3480	FB9E	8D0A02		STA LDR+3	
3490	FBA1	60		RTS	
3500	FBA2	A209	BLDEL	LDX #9	BLOCK DELETE ROUTINES
3510	FBA4	CA	MOVE	DEX	
3520	FBA5	A92F		LDA ##2F	
3530	FBA7	B413		LDY \$13,X	
3540	FBA9	943B		STY \$3B,X	
3550	FBA8	D53B		CMP \$3B,X	
3560	FBAD	1009		BPL DONE	
3570	FBAF	A939		LDA ##39	
3580	FBB1	D53B		CMP \$3B,X	
3590	FBB3	10EF		BPL MOVE	
3600	FBB5	4C3BFB	WRONG	JMP PRNT	
3610	FBB8	CA	DONE	DEX	
3620	FBB9	E003		CPX #3	
3630	FBBB	D0F8		BNE WRONG	
3640	FBBD	A9FB		LDA #VECTOR/256	
3650	FBBF	8D1902		STA \$219	
3660	FBC2	A9D8		LDA #VECTOR*256/256	
3670	FBC4	8D1802		STA \$218	
3680	FBC7	B513	LOAD	LDA \$13,X	
3690	FBC9	9D2802		STA LINUM,X	
3700	FBCB	CA		DEX	
3710	FBCD	10F8		BPL LOAD	
3720	FBCF	202EF9		JSR LINCO	
3730	FBD2	68		PLA	
3740	FBD3	A8		TAY	
3750	FBD4	68		PLA	
3760	FBD5	A90D	XTT	LDA ##D	
3770	FBD7	60		RTS	


```

3780 FBD8 A203 VECTOR LDX #3
3790 FBDA BD2802 DELL LDA LINUM,X
3800 FBDD D540 CMP #40,X
3810 FBDF F006 BEQ DELEND
3820 FBE1 202BF9 JSR LINC1
3830 FBE4 4CD5FB JMP XTT
3840 FBE7 CA DELEND DEX
3850 FBE8 10F0 BPL DELL
3860 FBEA A9F8 LDA #BEGIN/256
3870 FBEC 8D1902 STA #219
3880 FBEE A9E2 LDA #BEGIN*256/256
3890 FBF1 8D1802 STA #218
3900 FBF4 8D3102 STA BLFLAG
3910 FBF7 D0DC BNE XTT
3920 ;END OF IT ALL!

```

3780 FBD8 A203
3790 FBDA BD2802
3800 FBDD D540
3810 FBDF F006
3820 FBE1 202BF9
3830 FBE4 4CD5FB
3840 FBE7 CA
3850 FBE8 10F0
3860 FBEA A9F8
3870 FBEC 8D1902
3880 FBEE A9E2
3890 FBF1 8D1802
3900 FBF4 8D3102
3910 FBF7 D0DC
3920