

```

10 0000 ;*****
20 0000 ; INPUT AND SCAN ROUTINES
30 0000 ; AUTHOR: J. A. HINDS
40 0000 ; COPYRIGHT NESTAR SYSTEMS
50 0000 ; 1979
60 0000 ;*****
70 0000 ; JUMP VECTORS:
80 700C *=RDLINE GET INPUT TEXT LINE
90 700C 4C7B70 JMP GIL FROM OPERATOR
100 7003 *=SCAN SCAN FOR LITERAL
110 7003 4C1E71 JMP SCLIT SPECIFIED IN-LINE
120 7006 *=NUMBER SCAN FOR INTEGER
130 7006 4CA170 JMP SINT PUT VALUE IN GPTR
140 701B *=RETURN RE-ENTER IN-LINE SCANNING ROUTINES
150 701B 4CE471 JMP REACT
160 700F *=PACK CRUNCH TEXT AT NXTCHR IN PLACE
170 700F 4C3B72 JMP PACKE
180 7009 *=CMDEND RETURN <>0 IF AT END OF COMMAND LINE
190 7009 4C5671 JMP SCEND
200 7012 *=UNPACK UNCRUNCH TEXT AT LPTR INTO LBUF
210 7012 4C1172 JMP UPACKE
220 7018 *=ER2ENT PRINT AN ERROR MESSAGE WITHOUT POINTER TO C
OMMAND LINE
230 7018 4CD171 JMP ERR2
240 7015 *=ERRENT PRINT AN ERROR MESSAGE WITH POINTER TO COMM
AND LINE
250 7015 4CB871 JMP ERR
260 701E *=PRINTA TRANSLATE CHAR FROM ASCII TO PET AND PRINT
270 701E 4C5B73 JMP PROUT
280 7021 ;
290 707B *=BSCAN
300 707B PRCHAR=$904B
310 707B CRCHAR =13 CARRAIGE RETURN
320 707B BLANK = $20 ASCII BLANK OR SPACE
330 707B ;*****
340 707B ;
350 707B ; GET INPUT LINE INTO INBUF
360 707B ; KILLS ALL REGISTERS
370 707B ; SETS NXTCHR TO POINT TO FIRST
380 707B ; CHAR OF INBUF
390 707B ;
400 707B 209870 GIL JSR RSTNC POINT TO FIRST CHAR OF BUFFER
410 707E 208B72 GNXT JSR RDSCX
420 7081 C9A0 CMP #$A0 SHIFTEDT SPACE
430 7083 D002 BNE TRIXI
440 7085 A920 LDA #BLANK
450 7087 208373 TRIXI JSR TRIN TRANSLATE IN
460 708A A200 LDX #0
470 708C 8127 STA (NXTCHR,X) PUT CHAR IN BUFFER
480 708E E627 INC NXTCHR
490 7090 D002 BNE GDTST STEP TO NEXT CHAR POSITION
500 7092 E628 INC NXTCHR+1
510 7094 C90D GDTST CMP #CRCHAR
520 7096 D0E6 BNE GNXT IF NOT GET MORE
530 7098 A9D3 RSTNC LDA #INBUF*256/256
540 709A 8527 STA NXTCHR
550 709C A902 LDA #INBUF/256
560 709E 8528 STA NXTCHR+1
570 70A0 60 RTS
580 70A1 ;*****

```

```

600 70A1 ; SCAN FOR INTEGER
610 70A1 ; KILLS ALL REGISTERS
620 70A1 ; ZERO FLAG SET IF NO INTEGER
630 70A1 ; VALUE RETURNED IN GPTR
640 70A1 ; JUMP TO LABEL OVFL0 IF NUMBER
650 70A1 ; IS GREATER THAN 64K
660 70A1 ;
670 70A1 ;*****
680 70A1 206571 SINT JSR SUPRS IGNORE LEADING BLANKS
690 70A4 20C770 JSR VFY09 IS IT A NUMBER?
700 70A7 B003 BCS ISNUM CARRY MEANS YES
710 70A9 A000 LDY #0 SET ZERO FOR FAILURE
720 70AB 60 RTS
730 70AC A200 ISNUM LDX #0 ZERO OUT ACCUMULATOR
740 70AE 8625 STX GPTR
750 70B0 8626 STX GPTR+1
760 70B2 20D870 CONT JSR TTAA ACCUMULATE EACH DIGIT
770 70B5 C8 INY STEP ALONG IN INPUT STRING
780 70B6 20C770 JSR VFY09 ARE WE AT END OF NUM?
790 70B9 B0F7 BCS CONT
800 70BB 18 BUMPNX CLC
810 70BC 98 TYA ADD Y TO NXTCHR AND SIGNAL
820 70BD 6527 ADC NXTCHR FAIL/SUCCESS
830 70BF 8527 STA NXTCHR
840 70C1 9002 BCC RTNX
850 70C3 E628 INC NXTCHR+1
860 70C5 98 RTNX TYA SET ZERO/NONZERO AS CONTENTS OF Y
870 70C6 60 RTS
880 70C7 ;
890 70C7 A939 VFY09 LDA #'9 VERIFY NEXT CHAR IS NUMERIC
900 70C9 48 PHA
910 70CA A92F LDA #'/' ('0-1)
920 70CC D127 CMP (NXTCHR),Y
930 70CE 68 PLA
940 70CF B005 BCS RNGBAD
950 70D1 D127 CMP (NXTCHR),Y
960 70D3 B127 LDA (NXTCHR),Y AS SERVICE TO CALLER
970 70D5 60 RTS
980 70D6 18 RNGBAD CLC
990 70D7 60 RTS
1000 70D8 ;
1010 70D8 48 TTAA PHA TEN TIMES AND ADD
1020 70D9 0625 ASL GPTR GPTR:=GPTR*10+DIGIT IN A
1030 70DB 2626 ROL GPTR+1
1040 70DD B029 BCS OVFL
1050 70DF A525 LDA GPTR
1060 70E1 A626 LDX GPTR+1
1070 70E3 0625 ASL GPTR
1080 70E5 2626 ROL GPTR+1
1090 70E7 B01F BCS OVFL
1100 70E9 0625 ASL GPTR
1110 70EB 2626 ROL GPTR+1
1120 70ED B019 BCS OVFL
1130 70EF 6525 ADC GPTR
1140 70F1 8525 STA GPTR
1150 70F3 8A TXA
1160 70F4 6526 ADC GPTR+1
1170 70F6 8526 STA GPTR+1
1180 70F8 B00E BCS OVFL
1190 70FA 68 PLA
1200 70FB 290F AND #50F GET RID OF ASCII ZONE
1210 70FD 6525 ADC GPTR
1220 70FF 8525 STA GPTR
1230 7101 9004 BCC DTAA
1240 7103 E626 INC GPTR+1

```

```

1260 7107 60      DTTAA  RTS
1270 7108 20BB70  OVFL   JSR BUMPX MAKE NXTCHR POINT
1280 710B          ;           TO OFFENDING POSITION
1290 710B 201570  JSR ERRENT REPORT ERR WITH IN-LINE TEXT
1300 710E 4E      .BYTE 'NUMBER GT'
1300 710F 55
1300 7110 4D
1300 7111 42
1300 7112 45
1300 7113 52
1300 7114 20
1300 7115 47
1300 7116 54
1310 7117 20      .BYTE ' 65535', $00
1310 7118 36
1310 7119 35
1310 711A 35
1310 711B 33
1310 711C 35
1310 711D 00
1320 711E          ;*****
1330 711E          ;
1340 711E          ;   SCAN FOR LITERAL
1350 711E          ;   BLANK SUPPRESS INBUFF
1360 711E          ;   BEFORE COMPARE
1370 711E          ;   RETURNS NON-ZERO STATUS FOR SUCCESS
1380 711E          ;   WILL UPDATE NXTCHR ON SUCCESS
1390 711E          ;
1400 711E          ;   CLOBBERS ALL REGISTERS
1410 711E          ;
1420 711E          ;*****
1430 711E 20EE71  SCLIT  JSR ILTXT SET IPTR,GPTR,X TO IN-LINE TEXT
1440 7121 206571  SUPP   JSR SUPRS
1450 7124 206F71  SCANZ  JSR FILE   CHAR FROM CALLER
1460 7127 F015    BEQ DONE   END
1470 7129 D127    CMP (NXTCHR),Y COMPARE WITH USERS INPUT LINE
1480 712B F00E    BEQ ISMTCH OK!
1490 712D C941    CMP #'A
1500 712F 901D    BCC NOMTCH CONVERT CASE IF
1510 7131 C95B    CMP #1+'Z IN ALPHABET
1520 7133 B019    BCS NOMTCH
1530 7135 0920    ORA # $20
1540 7137 D127    CMP (NXTCHR),Y
1550 7139 D013    BNE NOMTCH
1560 713B C8      ISMTCH  INY   STEP ALONG INPUT BUFFER
1570 713C D0E6    BNE SCANZ (ASSUME THAT BRANCH IS ALWAYS TAKEN)
1580 713E C90D    DONE   CMP #CRCHAR IS STRING SUPPOSED TO END IN DELIMITER?
1590 7140 D00E    BNE DONE1 NO
1600 7142 B127    LDA (NXTCHR),Y USERS INPUT MUST END
1610 7144 29DF    AND # $DF IGNORE CASE
1620 7146 C941    CMP #'A WITH DELIMITER
1630 7148 9006    BCC DONE1 YES
1640 714A C95B    CMP #1+'Z
1650 714C B002    BCS DONE1 YES
1660 714E A000    NOMTCH LDY #0 SET FAILURE
1670 7150 20BB70  DONE1  JSR BUMPX
1680 7153 4C7E71  JMP ILTCLO GO BACK TO CALLER
1690 7156 206571  SCEND  JSR SUPRS
1700 7159 B127    LDA (NXTCHR),Y
1710 715B C90D    CMP #CRCHAR
1720 715D F003    BEQ ENDOK
1730 715F A900    LDA #0 SIGNAL FAIL
1740 7161 60      RTS
1750 7162 A901    ENDOK  LDA #1
1760 7164 60      RTS

```

```

1780 7165 ;
1790 7165 ; MISC INTERNAL SUPPORT ROUTINES
1800 7165 ;
1810 7165 ;*****
1820 7165 ;
1830 7165 A0FF SUPRS LDY #$FF SUPPRESS BLANKS IN
1840 7167 A920 LDA #BLANK INPUT BUFFER
1850 7169 C8 SPLP INY
1860 716A D127 CMP (NXTCHR),Y
1870 716C F0FB BEQ SPLP
1880 716E 60 RTS
1890 716F ;*****
1900 716F ;
1910 716F ; GET NEXT BYTE OF IN-LINE
1920 716F ; CHARACTER STRING
1930 716F ; FILB
1940 716F ; PROTOCOL: INITIALIZE WITH
1950 716F ; ILTXT
1960 716F ; FETCH EACH CHAR WITH
1970 716F ; FILB (ZERO SET AT END OF STRING)
1980 716F A123 FILB LDA (IPTR,X) GET IN-LINE BYTE
1990 7171 F00A BEQ FILBR SET ZERO IF END OF IT
2000 7173 C90D CMP #CRCHAR WITH ZERO OR CR
2010 7175 F006 BEQ FILBR DONT GO PAST END
2020 7177 F623 INC IPTR,X OTHERWISE STEP
2030 7179 D002 BNE FILBR ALONG IN-LINE
2040 717B F624 INC IPTR+1,X (ASSERT CANT SET ZERO)
2050 717D 60 FILBR RTS
2060 717E ;*****
2070 717E ;
2080 717E ; IN-LINE PARAMETER SUPPORT
2090 717E ;
2100 717E 8A ILTCLO TXA IS TEXT OUT OF LINE?
2110 717F D005 BNE RTNR YES DONT NEED TO BUMP INSTRUCTION POINTER
2120 7181 206F71 ILTLP JSR FILB
2130 7184 D0FB BNE ILTLP
2140 7186 A524 RTNR LDA IPTR+1
2150 7188 48 PHA USE IPTR AS RETURN ADDRESS
2160 7189 A523 LDA IPTR
2170 718B 48 PHA
2180 718C 98 TYA SET SUCCESS/FAIL AS Y REG
2190 718D 60 RTS
2200 718E ;
2210 718E 209D71 GETONE JSR GETBYT GET SPECIFIER BYTE
2220 7191 C902 GETILP CMP #2
2230 7193 F0F1 BEQ RTNR IN-LINE PROCEDURE
2240 7195 C901 CMP #1 INDIRECT ADDRESS?
2250 7197 F00D BEQ FINDR
2260 7199 209D71 JSR GETBYT DIRECT
2270 719C AA TAX LOW BYTE TO X
2280 719D E623 GETBYT INC IPTR
2290 719F D002 BNE FBYTE
2300 71A1 E624 INC IPTR+1
2310 71A3 B123 FBYTE LDA (IPTR),Y FETCH BYTE
2320 71A5 60 RTS
2330 71A6 209D71 FINDR JSR GETBYT
2340 71A9 8525 STA GPPTR
2350 71AB 209D71 JSR GETBYT
2360 71AE 8526 STA GPPTR+1
2370 71B0 B125 LDA (GPPTR),Y
2380 71B2 AA TAX LOW BYTE
2390 71B3 C8 INY
2400 71B4 B125 LDA (GPPTR),Y HIGH BYTE
2410 71B6 88 DEY
2420 71B7 60 RTS
2430 71B8 ;*****

```

```

2440 71B8 ; ERROR HANDLING ENTRY POINTS
2450 71B8 ;
2460 71B8 ; ERRENT: POINT TO OFFENDING POSITION
2470 71B8 ; (ERR) OF COMMAND LINE AND FALL INTO
2480 71B8 ; ERZENT: PRINT IN-LINE STRING OF ERROR
2490 71B8 ; (ERR2) MESSAGE, RESET STACK AND
2500 71B8 ; JUMP TO GLOBAL RESET LOCATION
2510 71B8 ;
2520 71B8 ;
2530 71B8 A90D ERR LDA #CRCHAR
2540 71BA 204B90 JSR PRCHAR
2550 71BD A92D LDA #'- PRINT ERR MESSAGE
2560 71BF A427 LDY NXTCHR
2570 71C1 C0D3 ERLP CPY #INBUF*256/256
2580 71C3 F007 BEQ ERLPX
2590 71C5 204B90 JSR PRCHAR
2600 71C8 88 DEY
2610 71C9 4CC171 JMP ERLP
2620 71CC A93F ERLPX LDA #'?
2630 71CE 204B90 JSR PRCHAR
2640 71D1 20EE71 ERR2 JSR ILTXT ALTERNATE ENTRY POINT
2650 71D4 A90D LDA #CRCHAR PREFACE WITH CR
2660 71D6 204B90 ERPLP JSR PRCHAR AND PRT TO
2670 71D9 206F71 JSR FILB IN LINE MSG
2680 71DC D0F8 BNE ERPLP ENDING
2690 71DE A2FE LDX #FE RESET STACK POINTER
2700 71E0 9A TXS
2710 71E1 4C2170 JMP COMMAN
2720 71E4 68 REACT PLA REENTER FROM IN-LINE PARAMETRIC PROCEDURE
2730 71E5 8523 STA IPTR
2740 71E7 68 PLA
2750 71E8 8524 STA IPTR+1
2760 71EA 98 TYA HIGH BYTE TO A, LOW IN X
2770 71EB A000 LDY #0
2780 71ED 60 RTS
2790 71EE ;*****
2800 71EE ;
2810 71EE ; OPEN IN-LINE TEXT
2820 71EE ;
2830 71EE ; ILTXT
2840 71EE ;
2850 71EE ; INITIALIZES IPTR,GPTR,X REG
2860 71EE ; TO ALLOW ACCESS TO TEXT STRING
2870 71EE ; BY CALLING FILB.
2880 71EE 68 ILTXT PLA LOCAL RETURN ADDR
2890 71EF AA TAX
2900 71F0 68 PLA
2910 71F1 A8 TAY
2920 71F2 68 PLA
2930 71F3 8523 STA IPTR CALLERS IN-LINE PARM
2940 71F5 68 PLA ADDRESS
2950 71F6 8524 STA IPTR+1
2960 71F8 98 TYA PUSH BACK LOCAL ADDRESS
2970 71F9 48 PHA
2980 71FA 8A TXA
2990 71FB 48 PHA
3000 71FC A000 LDY #0 FOR GETBYT
3010 71FE 209D71 JSR GETBYT
3020 7201 A200 LDX #0 ASSUME TEXT IN-LINE
3030 7203 C90A CMP #10
3040 7205 B009 BCS ILTXTX DONE
3050 7207 209171 JSR GETILP GET ADDRESS
3060 720A 8625 STX GPTR LOW BYTE
3070 720C 8526 STA GPTR+1
3080 720E A202 LDX #GPTR-IPTR
3090 7210 68 ILTXTX PLA IN-LINE TEXT EXIT

```

```

3100 7211 ;*****
3110 7211 ;
3120 7211 ; PACK/UNPACK ROUTINES
3130 7211 ; PACKE PACK TEXT INPLACE AT
3140 7211 ; NXTCHR LEAVE COUNT IN Y REG
3150 7211 ;
3160 7211 ; UPACKE UNPACK TEXT AT LPTR
3170 7211 ; INTO LBUF
3180 7211 ; CLOBBERS ALL REGISTERS
3190 7211 A002 UPACKE LDY #2 UNPACK TEXT
3200 7213 A200 LDX #0 FROM Y TO X
3210 7215 8625 STX GPPTR REPEAT COUNT
3220 7217 B131 UPL LDA (LPTR),Y
3230 7219 1006 BPL STIKIT
3240 721B 8525 STA GPPTR
3250 721D 88 DEY
3260 721E B131 LDA (LPTR),Y
3270 7220 C8 INY
3280 7221 9D8202 STIKIT STA LBUF,X
3290 7224 E8 INX
3300 7225 E050 CPX #80
3310 7227 F043 BEQ UPERR
3320 7229 E625 INC GPPTR
3330 722B 30F4 BMI STIKIT
3340 722D C625 DEC GPPTR DONT GET BIG!
3350 722F C90D CMP #CRCHAR
3360 7231 D001 BNE UPM
3370 7233 60 RTS
3380 7234 C8 UPM INY
3390 7235 C052 CPY #82
3400 7237 D0DE BNE UPL
3410 7239 F031 BEQ UPERR
3420 723B ;
3430 723B ; PACK LINE (IN-PLACE) AT LOCATION NXTCHR TO END OF LINE
3440 723B ;
3450 723B ; TEXT COMPRESSION PLACES A REPEAT COUNT
3460 723B ; AFTER A CHARACTER THAT IS
3470 723B ; REPEATED. THE REPEAT COUNT
3480 723B ; APPEARS AS A NEGATIVE BYTE
3490 723B ; (IN RANGE OF $80 TO $FF)
3500 723B ; THIS GIVES THE FOLLOWING
3510 723B ; EQUIVALENCES:
3520 723B ; SOURCE COMPRESSED
3530 723B ; ABBC AB(FF)C
3540 723B ; ABBBC AB(FE)C
3550 723B ; (THE SYMBOL(FE) INDICATES
3560 723B ; SINGLE BYTE VALUE IN THIS POSITION)
3570 723B ;
3580 723B ;
3590 723B ; THE COUNT OF CHARS IN THE COMPRESSED
3600 723B ; TEXT IS RETURNED IN THE Y REG
3610 723B A000 PACKE LDY #0
3620 723D 98 TYA
3630 723E AA TAX
3640 723F 8525 STA GPPTR
3650 7241 8526 STA GPPTR+1
3660 7243 B127 LDA (NXTCHR),Y
3670 7245 C90D PKCM CMP #CRCHAR DONE?
3680 7247 D003 BNE PKLP NO
3690 7249 C8 INY RETURN COUNT IN A
3700 724A 98 TYA
3710 724B 60 RTS
3720 724C CA PKLP DEX
3730 724D E626 INC GPPTR+1 "FROM" POINTER
3740 724F A426 LDY GPPTR+1
3750 7251 C051 CPY #81

```

```

3760 7253 B017      BCS UPERR TOO BIG LINE
3770 7255 B127      LDA (NXTCHR),Y
3780 7257 A425      LDY GPPTR
3790 7259 D127      CMP (NXTCHR),Y A DUPLICATE?
3800 725B F0EF      BEQ PKLP UPDATE REPEATE CNT
3810 725D E8        INX IS RPT CNT NULL?
3820 725E F005      BEQ PKST IF NULL STORE CHAR
3830 7260 8A        TXA OTHERWISE STORE COUNT
3840 7261 A200      LDX #0
3850 7263 C626      DEC GPPTR+1 AND RESCAN SOURCE
3860 7265 C8        PKST INY
3870 7266 8425      STY GPPTR
3880 7268 9127      STA (NXTCHR),Y
3890 726A D0D9      BNE PKCM
3900 726C 201570    UPERR JSR ERRENT
3910 726F 50        .BYTE 'PACK/UNPACK LINE LENGTH >80',500
3910 7270 41
3910 7271 43
3910 7272 4B
3910 7273 2F
3910 7274 55
3910 7275 4E
3910 7276 50
3910 7277 41
3910 7278 43
3910 7279 4B
3910 727A 20
3910 727B 4C
3910 727C 49
3910 727D 4E
3910 727E 45
3910 727F 20
3910 7280 4C
3910 7281 45
3910 7282 4E
3910 7283 47
3910 7284 54
3910 7285 48
3910 7286 20
3910 7287 3E
3910 7288 38
3910 7289 30
3910 728A 00

```

```

3920 728B ;*****
3930 728B ;
3940 728B ; SCREEN READ SURROGATE
3950 728B ; FOR TEXT APPLICATIONS
3960 728B ;
3970 728B ;*****
3980 728B ;
3990 728B ; PET SPECIFIC LOCATIONS
4000 728B LINEST =$E0 PTR TO SCRN LINE
4010 728B CURRCP =$E2 CURSOR POS IN LINE
4020 728B LASTCP =$F2 LAST POS OF LINE
4030 728B FLASH =$224 CURSOR VIS IF<>0
4040 728B CRSRON =$227 IS CURSON ON A CHAR
4050 728B OLDCHA =$226 IF SO THIS IS IT
4060 728B LINSRT =$220 PROMPT END OF LINE
4070 728B QUOTMD =$EA IN QUOTE MODE
4080 728B INSTMD =$FB IN INSERT MODE
4090 728B KBCNT =$20D NUM OF CHARS IN KBDBUF
4100 728B RINPRG =$260 IS READ IN PROGRESS
4110 728B ;
4120 728B GETKBB =$E27D GET CHAR FROM KBD BUF
4130 728B PRTPCHA =$E3EA A REG TO TV SCREEN
4140 728B

```

```

4150 728B 98      RDLNZZ = $E2CC SET PARMS FOR SCR READ
4160 728B        CONTRD = $E303 CONTINUE WITH PREVIOUS READ
4170 728B        ;
4180 728B 98      RDSCX TYA   SAVE REGS
4190 728C 48      PHA
4200 728D 8A      TXA
4210 728E 48      PHA
4220 728F AD6002   LDA RINPRG   IS A READ STILL IN PROGRESS
4230 7292 F009    BEQ CNTURD
4240 7294 4C03E3  JMP CONTRD
4250 7297 206173  ECHO   JSR TRCASE
4260 729A 20EAE3  JSR PRTCHA  CHAR TO TV, MOVE CURSOR
4270 729D A900    CNTURD LDA #0  TURN OFF QUOTE, INSERT MODES
4280 729F 85FB    STA INSTMD
4290 72A1 85EA    STA QUOTMD
4300 72A3 8D6803  STA CCSTAT
4310 72A6 AD0D02  IDLE2  LDA KBCNT
4320 72A9 8D2402  STA FLASH  LET CURSOR FLASH
4330 72AC F0F8    BEQ IDLE2   WAIT TILL AT LEAST ONE CHAR IN BUFFER
4340 72AE 78      SEI   DONT LET CURSOR FLASH FOR A WHILE
4350 72AF AD2702  LDA CRSRON IS CURSOR DESTROYING A CHAR
4360 72B2 F00B    BEQ GETKY  NO
4370 72B4 AD2602  LDA OLDCHA YES REPLACE IT ON TV
4380 72B7 A000    LDY #0
4390 72B9 8C2702  STY CRSRON
4400 72BC 20ACE7  JSR UPDSCR
4410 72BF 207DE2  GETKY  JSR GETKBB GET NEW CHARACTER
4420 72C2 C90D    CMP #CRCHAR
4430 72C4 D006    BNE PSPEC  TRY OTHER SPECIAL CHARS
4440 72C6 6E2002  ROR LINSRT SET CARRY IN THIS FLAG
4450 72C9 4CCCE2  JMP RDLNZZ
4460 72CC C912    PSPEC  CMP #18   RVS ON PET
4470 72CE F004    BEQ ISTAB
4480 72D0 C909    CMP #09  CONTROL I (TAB)
4490 72D2 D015    BNE ISHOME
4500 72D4 A0F7    ISTAB  LDY #256-9 SET CURSOR TO
4510 72D6 B97202  TABLP  LDA TABS-256+9, Y NEXT TAB
4520 72D9 C8      INY
4530 72DA D002    BNE TCMP
4540 72DC A9FF    LDA #$FF  GUARANTEED TO GET OUT
4550 72DE C5E2    TCMP   CMP CURRCP
4560 72E0 F0F4    BEQ TABLP STEP TO NEXT TAB
4570 72E2 90F2    BCC TABLP
4580 72E4 A8      TOUT  TAY   COUNT TO Y
4590 72E5 A200    LDX #0  CCSTAT BYTE
4600 72E7 F047    BEQ CHCH
4610 72E9        ;
4620 72E9 C913    ISHOME CMP #19   HOME KEY?
4630 72EB D01D    BNE ISCLR
4640 72ED 2E6803  ROL CCSTAT
4650 72F0 B006    BCS H2   IS SECOND HOME
4660 72F2 A000    LDY #0   FIRST OF LINE
4670 72F4 A280    LDX #$80 CCSTAT BYTE
4680 72F6 D038    BNE CHCH
4690 72F8 304E    H2     BMI H3   PLACE CURSOR AT BOTTOM LEFT CORNER
4700 72FA A4F2    LDY LASTCP
4710 72FC A920    LDA #BLANK
4720 72FE D1E0    TRIMH  CMP (LINEST), Y TRIM BLANKS
4730 7300 D003    BNE SLIME YES JUST PRINT IT
4740 7302 88      DEY
4750 7303 10F9    BPL TRIMH
4760 7305 C8      SLIME  INY
4770 7306 A2C0    LDX #$C0 SET SECOND TIME
4780 7308 D026    BNE CHCH
4790 730A C993    ISCLR  CMP #147 CLEAR KEY
4800 730C F000    BEQ CHCH

```



```

4810 730E 4C9772 RL2EC JMP ECHO
4820 7311 0E6803 CLXX ASL CCSTAT
4830 7314 0E6803 ASL CCSTAT
4840 7317 0E6803 ASL CCSTAT
4850 731A 2E6803 ROL CCSTAT LOOK AT $18 BITS
4860 731D B01F BCS CLR2
4870 731F A210 LDX # $10 FIRST TIME
4880 7321 A4E2 CLRF LDY CURRCP START AT THIS
4890 7323 A920 CRXX LDA #BLANK
4900 7325 91E0 CLRLP STA (LINEST),Y
4910 7327 C8 INY
4920 7328 C4F2 CPY LASTCP
4930 732A F0F9 BEQ CLRLP
4940 732C 90F7 BCC CLRLP
4950 732E A4E2 LDY CURRCP DONT UPDATE FOR CLEAR
4960 7330 C4F2 CHCH CPY LASTCP
4970 7332 9002 BCC SETCP DONT GO PAST END
4980 7334 A4F2 LDY LASTCP
4990 7336 84E2 SETCP STY CURRCP UPDATE CURSOR POSITION
5000 7338 8E6803 STSTR STX CCSTAT STATE FOR CURSOR CONTROLS
5010 733B 4CA672 JMP IDLE2
5020 733E 30CE CLR2 BMI RL2EC PRINT TO CLEAR SCREEN
5030 7340 A000 LDY #0
5040 7342 84E2 STY CURRCP
5050 7344 A218 LDX # $18 SET SECOND TIME
5060 7346 D0DB BNE CRXX STOW IT AWAY
5070 7348 2E6803 H3 ROL CCSTAT
5080 734B 30C1 BMI RL2EC 4TH TIME JUST PRINT
5090 734D A019 HBOT LDY #25
5100 734F 20EAE3 HBLP JSR PRTCHA
5110 7352 A911 LDA #17 CURSOR DOWN
5120 7354 88 DEY
5130 7355 D0F8 BNE HBLP
5140 7357 A2E0 LDX # $E0 LAST TIME
5150 7359 D0DD BNE STSTR
5160 735B 207973 PROUT JSR TROUT
5170 735E 4C4B90 JMP PRCHAR
5180 7361 ;*****
5190 7361 2C6703 TRCASE BIT CASE
5200 7364 1012 BPL NOCASE
5210 7366 C941 CMP # 'A
5220 7368 900E BCC NOCASE
5230 736A C95B CMP #1+ 'Z
5240 736C 9008 BCC DOCASE
5250 736E C9C1 CMP #193 SHIFTED A
5260 7370 9006 BCC NOCASE
5270 7372 C9DB CMP #219 SHIFTED Z +1
5280 7374 B002 BCS NOCASE
5290 7376 4980 DOCASE EOR # $80
5300 7378 60 NOCASE RTS
5310 7379 ;*****
5320 7379 C961 TROUT CMP #97 SMALL A
5330 737B 90FB BCC NOCASE
5340 737D C97B CMP #123 SMALL Z +1
5350 737F B0F7 BCS NOCASE
5360 7381 9008 BCC TOGGLE
5370 7383 ;
5380 7383 C9C1 TRIN CMP #193 SMALL A
5390 7385 90F1 BCC NOCASE
5400 7387 C9DB CMP #219 SMALL Z +1
5410 7389 B0ED BCS NOCASE
5420 738B 49A0 TOGGLE EOR # $A0
5430 738D 60 RTS
5440 738E ;*****

```