

TIOS 99

TI PUGET SOUND 99ERS

P.O. BOX 6073
LYNNWOOD, WA 98036

BULK RATE
FIRST CLASS PERMIT
PAID
Lynnwood, WA 98036
Permit No. 159

035
8706

EXTENDED BASIC PROGRAMMING

JUNE 1987 Vol. 6 No. 06
MEMBERSHIP \$15.00 PER YEAR NEWSLETTER PRICE (FOR NON-MEMBERS) \$1.50 PER ISSUE

OFFICERS

PRESIDENT : CHUCK WYNNE	745-3249	VICE PRESIDENT: GENE CHRISTIANSEN (Call Chuck Wynne)
SECRETARY : DENNIS WOOD	641-0298	TREASURER : DAVE POWELL 631-7794
NEWSLETTER: TOM WYNNE	745-3249	LIBRARIAN : JOHN RULAND 672-7593

NEXT MEETING

DATE : THURSDAY, July 23, 1987 TIME: 7:00 PM
PLACE: Kirkland Public Library, 406 Kirkland Ave, Kirkland

- | | |
|-----------------------------|-------------------------------|
| AGENDA: * Old business | * Extended BASIC tutorial |
| * Modem programs | * New software demonstrations |
| * Continuing GENEVE updates | * Question and Answer session |

AUGUST MEETING

DATE : Thursday, August 20, 1987
TIME : 7:00 PM
PLACE: Kirkland Public Library
406 Kirkland Ave, Kirkland

IN THIS NEWSLETTER

The President Says. 3

Programming the TI99/4A

Extended BASIC:

Changing your keyboard using CALL KEY	4
Fireworks	5
Pre-Formatter	6
Extended BASIC screen color change.	7
Cursor change	8
Wave power II	9
WHIP.	9
Dulcimer.	10
Word Scramble	11
Colorful Pattern.	11
Wee Bomber.	12
Gallery	13
Traffic Cop	14

SOME LOCAL BULLETIN BOARD NUMBERS

<u>TITLE</u>	<u>PHONE NUMBER</u>
PUGET SOUND 99ERS BBS-SEATTLE	784-4142
TIBBS NW-ED DURFFEE-BELLEVUE	641-5884
QUEEN ANNE COMPUTER SHOPPE	361-0895
NEW WORLD BBS	365-6938
BAINBRIDGE ISLAND	1-598-3228
TELETRAVEL	742-9034
SHUTTLE	885-INFO
TROTHGUARD-BURIEN	824-6757
RADIO 1	878-0158
Genie	1-800-638-8369

THE PRESIDENT SAYS

By Chuck Wynne

Want to buy a good modified console? Well, mine is not for sale, but just thought I'd try and show you how many modifications there are. I wonder if any other computer in the world has as many user changeable options in the works.

The first thing I did was install 32K right on the console itself. But wait, some have gone further. They have put much more memory than that on board. But can you access it? You bet you can. By bank switching (other computers do-it's in the software), or by manual switches on the chips. Oh, by the way, I didn't use the latest option for the 32K. Now instead of 4 chips, it involves only one!

I put a diode in series with the alpha lock key. Now when using joysticks, no need to worry about the alpha lock key. I also put a 390 ohm resistor in place of the 560 ohm VDP load resistor. It gave me some improvement on the video end. (See Page 8)

And how about a switch for the crystal. Makes the computer faster changing from a 12 MHTZ crystal to a 14.318 crystal. (see Computer Shopper, October 1986, p203) It can foul up the RS232 port with the faster speed though.

Some have put the module chips on board also. Here is a question. They have a no slot clock out there that they say works with any computer. It can fit into any RAM, or ROM slot. Can anyone tell me if it can be used with a TI?

Word is out about a new PE box. It should run around \$150.00, and will support the TI, or Geneve computer.

CorComp is supposedly producing a cartridge which will copy IBM to TI, or vice-versa on your two drives. Hmmm, sounds like something I would buy. But wait a minute. If it will do that, why not have that software part of the Geneve package, which will cause that computer to be able to read IBM formatted disks, or maybe even use them directly. It wouldn't be an IBM, but it could be better than an IBM by far.

Computer Shopper is still supporting us somewhat. Please, if you buy from this magazine, make sure they know you are a TI computer user. They think we don't even dent the market. I think we are a big dent!

Did you say there was something another computer could do that the TI can't? If you seem to think so, and someone in the TI community finds out about it, it seems to sometimes become possible. And if you support these individuals, they come through. Wish lists (with a little money) can make your wildest dreams come true.

If you are not communicating with a modem, chances are that you are not getting full use of your computer. The

means you can get program updates as well as new ones over the phone lines. It also means getting support from other TI users. Long distance you say-too expensive. Well for \$25 a month (off hours) you can modem the nation. No long distance charges! And that's a link that goes by satellite.

CHANGING YOUR KEYBOARD USING CALL KEY

By Tom Wynne

The following program will show how call key can be used to change the keyboard mode. It first allows you to type in upper and lower case but then changes the keyboard so you can only type upper case, regardless of the alpha lock being up or down. This is handy for prompting the user to enter a filename. It keeps the filename from being typed in lower case. TI-Writer, for example, allows you to type in lower case in the editor, but when you hit FCTN 9 to get to the menu line on top of the screen, you can only type in upper case.

```
100 PRINT "YOU CAN TYPE IN BOTH UPPER AND LOWER CASE"
110 INPUT "":A$
120 PRINT A$
130 CALL KEY(3,K,S)
140 PRINT "NOW YOU CAN ONLY TYPE IN UPPER CASE"
150 INPUT "":A$
160 PRINT A$
170 CALL KEY(5,K,S)
180 PRINT "YOU CAN TYPE BOTH AGAIN"
190 INPUT "":A$
200 PRINT A$
```

The next program will go through all the key scan modes. When you run this program, type and try to use the function keys, arrows, etc. and see what happens with each.

```
100 FOR I=0 TO 5
110 CALL KEY(I,K,S)
120 PRINT "TYPE SOMETHING"
130 INPUT "":A$
140 PRINT "YOU TYPED: ":A$
150 NEXT I
```

FIREWORKS

By Tom Wynne

Here is one for the season. Enjoy!!

```

100 : *****
110 ! *      FIREWORKS      *
120 ! *      BY: TOM WYNNE  *
130 ! * PUGET SOUND 99ERS *
140 ! *      JUNE 13, 1987  *
150 : *****
160 RANDOMIZE
170 DATA 00000000000000000000000000000000
      00000000000000000000000000000000
180 DATA 00000000000000010100000000000000
      00000000000000080800000000000000
190 DATA 00000000000003010102000000000000
      0000000000000C0C0C0C00000000000000
200 DATA 00000000000403010106010000000000
      0000000000040C0E0C0C02000000000000
210 DATA 00000000000C03010106090100000000
      000000002040C0F0C0C0201000000000
220 DATA 00000000100C03010106190102000000
      000000202040C0F0C0C0201008000000
230 DATA 00000020100C03010106192102040000
      000010202040CC70C0C0201008040000
240 DATA 00000060100C02000006192102040000
      0010102020420C3000C0201008040000
250 DATA 00000060100C00000004192102040000
      0010102020020C100000201008040000
260 DATA 00000060100800000000182102040000
      0010102000020C000000001008040000
270 DATA 00000060100000000000102002040000
      00101000000204000000000008040000
280 DATA 00000060000000000000002000040000
      00100000000200000000000000040000
290 DATA 00000040000000000000000000000000
      00000000000000000000000000000000
300 CALL CLEAR :: CALL SCREEN(2):: CALL MAGNIFY(4)
310 FOR I=0 TO 8 :: CALL COLOR(I,16,2):: NEXT I
      CALL COLOR(2,9,2)
320 FOR I=1 TO 6 :: CLR(I)=16 :: NEXT I
330 FOR I=1 TO 13 :: READ A$ :: CALL CHAR(88+I*4,A$)::
      NEXT I
340 CALL CHAR(42,"00071F3F7F7FFFFFFF",43,"00E0F8FCFFFFFFF")
350 CALL HCHAR(24,15,42):: CALL HCHAR(24,16,43)
360 DISPLAY AT(10,4):"HAPPY 4TH OF JULY!!!"
370 FOR I=4 TO 6 :: CALL SPRITE(#I,96,16,192,104,-RND*15+1,
      (RND*3-1)*RND*4):: NEXT I

```

```

380 CALL POSITION(#4,R,C):: CALL SPRITE(#1,92,16,R,C)::
    FOR I1=1 TO 4 :: CALL PATTERN(#1,88+I1*4):: NEXT I1
390 CALL POSITION(#5,R,C):: CALL SPRITE(#2,92,16,R,C)::
    FOR I2=1 TO 4 :: CALL PATTERN(#1,88+I1+I2)::
    CALL PATTERN(#2,88+I2):: NEXT I2
400 SS(1)=124 :: SS(2)=108 :: SS(3)=92
410 CALL POSITION(#6,R,C):: CALL SPRITE(#3,92,16,R,C)
420 FOR I=1 TO 3 :: CALL PATTERN(#I,SS(I)):: SS(I)=SS(I)+4
    :: IF SS(I)=144 THEN SS(I)=92 :: GOTO 430 ELSE 450
430 CALL POSITION(#I+3,R,C):: CALL SPRITE(#I,92,CLR(I+3),R,
    C):: CLR(I+3)=INT(RND*14)+3
440 CALL SPRITE(#I+3,96,CLR(I+3),192,104,-(RND*12+5),
    RND*10-5)
450 NEXT I :: GOTO 420

```

PRE-FORMATTER

By Tom Wynne

The following program will take a TI-WRITER file and put carriage returns at the end of each line. It also has the option of whether you want to replace all spaces with required spaces " ". If you want to replace anything else in the file, change line 200. C1\$ is the item you want to replace and C2\$ is the item you want to replace it with. For instance, line 200 could read: C1\$="&" :: C2\$="&&". And it would replace all '&' with '&&'. I created it for the use of entering program listings and fixed format files in a document for use with the Formatter of TI-Writer. When you run the program, it will ask you for a filename. Here, you enter the name of the file (DSK1.MYFILE) to insert carriage returns. If the file already has carriage returns, it will not put another in. It will only put a carriage return on the end of lines that do not have it. WARNING: LARGE FILES WILL CAUSE A MEMORY FULL ERROR.

```

100 ! *****
110 ! *   PRE-FORMATTER   *
120 ! * PUTS CR'S AT END *
130 ! * OF EACH LINE AND *
140 ! * REPLACES SPACES  *
150 ! * WITH REQUIRED     *
160 ! * SPACE.         *
170 ! *   BY: TOM WYNNE  *
180 ! *****
190 DIM A$(300)
200 C1$=" " :: C2$=" "
210 PRINT "ENTER FILE NAME:"
220 ACCEPT BEEP:FN$
230 PRINT "REPLACE "";C1$;"" WITH "";C2$;""?";

```

```

240 ACCEPT VALIDATE("YN"):YN$ :: IF YN$="" THEN 240
250 OPEN #1:FN$,INPUT
260 I=0
270 PRINT "READING FILE..."
280 IF EOF(1)THEN 320
290 LINPUT #1:B$
300 IF YN$="Y" THEN CALL REPLACE(B$,C1$,C2$)
310 A$(I)=B$ :: I=I+1 :: GOTO 280
320 CLOSE #1
330 PRINT "WRITING FILE..."
340 OPEN #1:FN$,OUTPUT
350 FOR J=0 TO I-1 :: IF A$(J)="" THEN 370
360 IF SEG$(A$(J),LEN(A$(J)),1)=CHR$(13)THEN 380
370 A$(J)=A$(J)&CHR$(13)
380 PRINT #1:A$(J):: NEXT J :: CLOSE #1
390 PRINT "FINISHED."
400 GOTO 210
410 SUB REPLACE(A$,C1$,C2$)
420 B$=""
430 FOR I=1 TO LEN(A$)
440 CH$=SEG$(A$,I,LEN(C1$))
450 IF CH$=C1$ THEN CH$=C2$
460 B$=B$&CH$ :: NEXT I :: A$=B$
470 SUBEND

```

EXTENDED BASIC SCREEN COLOR CHANGE

Changes the color of your screen while you program.

WARNING! Save this program before you run it!

```

100 ! PERMANENT SCREEN COLOR
110 ! CHANGE
120 ! FROM THE TI*MES
130 CALL CLEAR
140 INPUT "FOREGROUND COLOR? (1-16)":F
150 INPUT "BACKGROUND COLOR? (1-16)":B
160 C=16*(F-1)+(B-1)
170 CALL INIT :: CALL LOAD(9984,C,C,C,C,C,C,C,C,2,0,7,15+B,
4,32,32)
180 CALL LOAD(9999,48,2,0,8,0,2,1,39,0,2,2,0,8,4,32,32,36,
2,0,8,8,4)
190 CALL LOAD(10021,32,32,36,2,0,8,16,4,32,32,36,2,0,8,24,
4,32,32,36,4,91)
200 INPUT "IS THIS OK? (Y/N)":YN$
210 IF YN$="N" THEN 130
220 CALL LOAD(-31804,39,8)
230 CALL LOAD(-31952,255,231,255,231)
240 END

```

CURSOR CHANGE

By R.G. Albright

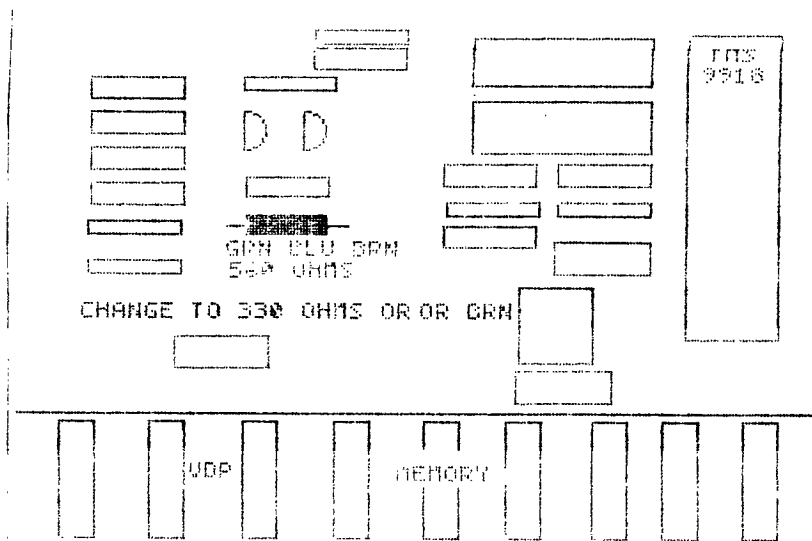
The following program will change the pattern of your cursor in Extended BASIC. This program will change it to the shape of Texas. If you want a different pattern, change line 140 to:

```
140 CALL LOAD(12288,X,X,X,X,X,X,X,X)
```

Where the X's represent different pattern definitions you can define. For example:

```
CALL LOAD(12288,0,0,0,0,0,0,0,0,252) underline cursor
CALL LOAD(12288,0,126,66,66,66,66,66,126) box cursor
```

```
100 ! TEXAS CURSOR BY DR. R.G. ALBRIGHT
110 CALL CLEAR :: CALL INIT
120 CALL LOAD(8196,63,248)
130 CALL LOAD(16376,67,85,82,83,79,82,48,8)
140 CALL LOAD(12288,48,48,63,255,254,124,24,12)
150 CALL LOAD(12296,2,0,3,240,2,1,48,0,2,2,0,8,4,32,32,36,
    4,91)
160 CALL LINK("CURSOR")
170 INPUT X$ :: GOTO 170
```



SAVE POWER II

```

100 !*****
110 !*      WAVE POWER II      *
120 !*      by                *
130 !*      JOHN MARTIN      *
140 !*      Southern Nevada   *
150 !*      Users Group      *
160 !*                        *
170 !* BASED ON A PROGRAM    *
180 !*      by                *
190 !*      JOHN WILLFORTH   *
200 !*      WEST PENN 99'ers *
210 !*                        *
220 !*      "WAVES IN COLOR" *
230 !*****
240 CALL CLEAR :: CALL SCREEN(2):: S=3 :: DIM Q$(12)::
GOTO 260 :: A$,B$,C$,D$,E$,F$,G$,H$,X,N,L :: CALL CHAR
:: CALL COLOR
250 !@P-
260 FOR X=1 TO 7 :: CALL CHAR(X+95,RPT$("0",14-X*2)&"FFFF")
:: NEXT X
270 H$="`abcdefedcba`ab" :: FOR X=1 TO 12 ::
Q$(X)=SEG$(H$,X,LEN(H$)-X-2)&H$&SEG$(H$,4,X):: NEXT X
280 FOR N=1 TO 2 :: L=ABS((N-2)*12):: FOR X=1 TO 12 ::
DISPLAY AT(X+L,1):Q$(X):: NEXT X :: NEXT N ::
PRINT Q$(12)
290 CALL COLOR(9,S,2):: FOR N=1 TO 8 :: FOR X=1 TO 12 ::
PRINT Q$(X):: NEXT X :: NEXT N
300 FOR N=1 TO 8 :: FOR X=12 TO 1 STEP -1 :: PRINT Q$(X)::
NEXT X :: NEXT N
310 FOR N=1 TO 8 :: FOR X=2 TO 9 :: PRINT Q$(X):: NEXT X ::
FOR X=8 TO 1 STEP -1 :: PRINT Q$(X):: NEXT X :: NEXT N
320 S=S+2 :: IF S>16 THEN S=3 :: GOTO 290 ELSE 290

```

WHIP

By R. Lilley

```

10 ! *****
20 ! *      WHIP              *
30 ! *      BY R. LILLEY     *
40 ! * SIDNEY USERS' GROUP  *
50 ! *      AUSTRALIA       *
60 ! *****
100 CALL CLEAR :: CALL CHAR(96,"3C7EFFFFFFF7E3C"):: J=-1
110 FOR L=1 TO 28 :: CALL SPRITE(#L,96,5,L*4,10,0,L*1)::
NEXT L
120 FOR L=1 TO 28 :: CALL MOTION(#L,0,L*J):: NEXT L
130 J=J*-1 :: GOTO 120

```

DULCIMER

Reprinted from the LA 99ers

Programed by: Jim Peterson

The accepted definition of a TINYGRAM is a complete program which can be listed on one screen. This one by Jim Peterson is a good example. I found it in the Dallas 99 Interface. Quote: "Type it in and you'll hear a hammered-dulcimer version of 'The Wildwood Flower'."

```

100 ! *****DULCIMER*****
110 ! *****A TINYGRAM*****
120 ! ***BY JIM PETERSON***
130 CALL CLEAR :: DIM S(26):: F=262 :: FOR N=1 TO 25 ::
      S(N)=INT(F*1.059463094^(N-1)):: NEXT N :: READ N ::
      C=S(N):: D=S(N)
140 RESTORE 180 :: FOR J=1 TO 63 :: GOSUB 160 :: NEXT J
150 U=U+2 :: CALL SOUND(-200,S(N),U,C,U,D,U)::
      IF U>27 THEN U=0 :: GOTO 140 ELSE 150
160 READ N :: CALL SOUND(-500,S(N),0)::
      CALL SOUND(-500,S(N),0,C,9)::
      CALL SOUND(-500,S(N),0,C,9,D,19):: D=C
170 C=S(N):: RETURN
180 DATA 5,6,8,8,10,13,5,5,6,5,3,3,5,3,1,1
190 DATA 5,6,8,8,10,13,5,5,6,5,3,3,5,3,1,1
200 DATA 8,13,17,17,17,15,13,13,8,8,10,10,13,10,8,8
210 DATA 1,1,1,3,5,5,8,5,3,3,5,3,1,1,1

```

But that's not all! Want to hear another tune? This program puts a two octave scale in a 25 element DIMension. Take the tune you want to hear and break it down into its respective parts (Ex, C, B#, etc.) and then use the chart below to match it to its DIMensional value:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
C	C#	D	D#	E	F	F#	G	G#	A	A#	B	C	C#	D
16	17	18	19	20	21	22	23	24	25					
D#	E	F	F#	G	G#	A	A#	B	C					

Put these numbers into the DATA statements. Each number is approximately one quarter long. To make the note longer, play it twice. Finally, count all your numbers and alter the FOR-TO statement in line 140.

WORD SCRAMBLE

From the Chicago Times Newsletter

```

100 ! WORD SCRAMBLE
110 ! CHICAGO TIMES
120 ! APRIL, 1987
130 CALL CLEAR
140 RANDOMIZE
150 INPUT "WORD TO SCRAMBLE: ":W$
160 FOR Y=1 TO 10
170 FOR I=1 TO LEN(W$)
180 R=INT(RND*LEN(W$)+1)
190 IF A(R) THEN 180
200 A(R)=I
210 X$=X$&SEG$(W$,R,1)
220 NEXT I
230 N$(Y)=X$ :: X$=""
240 PRINT N$(Y); " ";
250 FOR I=1 TO LEN(W$)
260 A(I)=0 :: NEXT I
270 NEXT Y
280 INPUT "REPEAT THAT WORD Y/N ":AN$
290 IF AN$="Y" THEN 160
300 STOP
300 CALL DELSPRITE(#3,#1):: SCORE=SCORE+1

```

COLORFUL PATTERN

```

10 ! MICROPENDIUM COLOR GRAPHICS
20 CALL CLEAR :: CALL INIT :: CALL LOAD(-31788,232)
30 PRINT "PRESS ENTER"
40 CALL KEY(0,K,S):: IF S=0 THEN 40 ELSE CALL SCREEN(2)
50 CALL CLEAR :: FOR T=1 TO 7 :: FOR I=34 TO 126
60 PRINT CHR$(I);:: NEXT I :: NEXT T
70 FOR I=34 TO 126 :: CALL VCHAR(1,1,I,768):: NEXT I ::
  GOTO 50

```

WEE BOMBER

By J. Pierre Pelletier

```
1 !WEE BOMBER BY J. PIERRE PELLETIER
100 CALL CLEAR :: DISPLAY AT(8,12):"THE": :TAB(6);
    "<< WEE BOMBER >>"
110 N=0 :: FOR I=1 TO 600 :: NEXT I
120 DISPLAY AT(20,2):"PRESS SPACE BAR TO FIRE":
    "'Q' WHEN TARGETS ARE GONE"
130 DISPLAY AT(24,1):"PRESS <ENTER> WHEN READY"
140 CALL KEY(0,K,S):: IF S=0 THEN 140
150 CALL CLEAR :: CALL HCHAR(22,2,30,30)
160 CALL COLOR(9,4,4):: CALL HCHAR(23,1,96,32)
170 CALL SPRITE(#1,43,2,35,256,0,-10)
180 CALL KEY(0,K,S):: CALL SOUND(-2000,-3,3,200,10)
190 IF S=0 THEN 180
200 IF K=81 THEN 320
210 CALL POSITION(#1,XX,YY):: AA=XX
220 CALL SPRITE(#2,46,2,XX,YY,20,0)
230 N=N+1 :: DISPLAY AT(24,2):"BOMBS:";N
240 AA=AA+4 :: CALL LOCATE(#2,AA,YY)
250 IF AA>=21*8 THEN CALL SOUND(100,-7,0)
260 CALL POSITION(#2,R,S)
270 FOR I=1 TO 375 :: NEXT I
280 CALL HCHAR(22,S/8+1,32):: AA=AA+S/8+1 ::
    CALL SOUND(-600,110,3,-7,0)
290 CALL DELSPRITE(#2):: GOTO 180
300 CALL SOUND(-100,-3,0,2000-(AA*4),10)
310 CALL DELSPRITE(#2):: GOTO 240
320 CALL DELSPRITE(ALL):: DISPLAY AT(24,1):" "
330 DISPLAY AT(11,7):"YOU USED";N;"BOMBS"
340 IF N=30 THEN 360
350 IF N>35 THEN 380 ELSE 390
360 DISPLAY AT(12,7):"YOUR AIM WAS PERFECT"
370 GOTO 410
380 DISPLAY AT(12,7):"YOUR AIM WAS FAIR"
390 GOTO 410
400 DISPLAY AT(12,7):"YOUR AIM WAS VERY GOOD"
410 FOR I=1 TO 600 :: NEXT I
420 GOTO 100
```

GALLERY

By R.W. August

```
100 ! GALLERY BY R.W. AUGUST
110 ! FROM BUG NEWS MAY 1987
120 CALL CLEAR :: CALL SCREEN(15)
130 CALL CHAR(36,"0E16FF7F1F0701"&RPT$("0",21)&
"1FFDFEDED8F"&RPT$("0",17))
140 CALL CHAR(40,"0E16FF7F1F0701"&RPT$("0",21)&
"1FF1FD3BE18F"&RPT$("0",17))
150 CALL CHAR(44,"0101"&RPT$("0",59))
160 CALL CHAR(64,"FFFFFFFFFFFFFFFF")
170 CALL CHAR(96,"181818183C3C3C3C")
180 CALL CHAR(124,"CC66FFFFFFFFFFFF")
190 CALL COLOR(12,5,1):: CALL MAGNIFY(4)
200 CALL HCHAR(4,1,124,32)
210 CALL HCHAR(1,1,64,96)
220 CALL HCHAR(24,16,96)
230 SCORE=0 :: DISPLAY AT(10,1):"SCORE";SCORE
240 CALL SPRITE(#1,36,16,10,80,0,-20,#2,40,16,9,200,0,-20)
250 DISPLAY AT(21,1):"Press:: : "<SPACE BAR>" ::
DISPLAY AT(24,1)SIZE(12):" To hoot"
260 CALL KEY(0,K,S):: IF K<>32 THEN 260
270 CALL SPRITE(#3,44,7,182,110,-90,0)
280 CALL COINC(#1,#3,18,A):: IF A THEN 300
290 CALL COINC(#3,1,110,15,B):: IF B THEN 340 ELSE 280
300 CALL DELSPRITE(#3,#1):: SCORE=SCORE+1
310 DISPLAY AT(10,6):SCORE
320 CALL SOUND(500,700,0)
330 CALL LOCATE(#1,140,10):: GOTO 360
340 CALL DELSPRITE(#3):: GOTO 260
350 DISPLAY AT(23,1):"<ENTER> or" :: DISPLAY AT(24,1)
SIZE(12):"S to reStart"
360 CALL KEY(0,K,S):: IF S=0 THEN 360
370 IF K=32 THEN 400
380 IF K=13 THEN 240
390 IF K<>83 THEN 360 ELSE 230
400 CALL SPRITE(#3,44,7,182,110,-90,0)
410 CALL COINC(#2,#3,18,A):: IF A THEN 430
420 CALL COINC(#3,1,110,20,B):: IF B THEN 470 ELSE 410
430 CALL DELSPRITE(#2,#3):: SCORE=SCORE+1
440 DISPLAY AT(10,6):SCORE
450 CALL SOUND(500,700,0)
460 CALL LOCATE(#2,140,50):: GOTO 350
470 CALL DELSPRITE(#3):: GOTO 360
480 END
```

TRAFFIC COP

From Tid-Bits of Australia

```
100 REM *****
110 REM *   TRAFFIC COP   *
120 REM *   FROM TID-BITS *
130 REM *   AUSTRALIA   *
140 REM *****
150 CALL CLEAR
160 PRINT "* TRAFFIC COP *"
170 REM
180 CALL MAGNIFY(3)
190 CALL GRAPHICS
200 CALL INSTRUCTIONS
210 CALL CLEAR
220 SP=8 :: CR=0
230 CALL SETUP(SP)
240 CALL CARS(SP)
250 CALL COINC(ALL,H):: IF H THEN CALL CRASH(CR)ELSE 280
260 CALL SOUND(-100,1000,5)
270 IF CR>9 THEN 350 ELSE 240
280 CALL KEY(3,K,S)
290 CT=CT+1 :: IF CT>75 THEN CALL FASTER(CT,SP,BST)
300 IF K=83 THEN CALL MOTION(#1,0,0)ELSE
    CALL MOTION(#1,0,SP)
310 IF K=68 THEN CALL MOTION(#2,0,0)ELSE
    CALL MOTION(#2,0,-SP)
320 IF K=69 THEN CALL MOTION(#3,0,0)ELSE
    CALL MOTION(#3,SP,0)
330 IF K=88 THEN CALL MOTION(#4,0,0)ELSE
    CALL MOTION(#4,-SP,0)
340 GOTO 250
350 CALL CLEAR
360 DISPLAY AT(6,1):"YOU CAUSED 10 CRASHES, WITH"
370 DISPLAY AT(8,1):"THE TRAFFIC RUNNING AT"
380 DISPLAY AT(10,1):SP;"MPH"
390 DISPLAY AT(12,2):"BEST SPEED SO FAR IS";BST;"MPH"
400 DISPLAY AT(16,6)BEEP:"PLAY AGAIN? (Y/N)"
410 CALL KEY(3,K,S):: IF S=0 THEN 410
420 IF K=78 THEN END
430 IF K<>89 THEN 400
440 CALL CLEAR :: GOTO 200
450 SUB GRAPHICS
460 CALL SCREEN(2)
470 CALL CHAR(128,"FF101010101010FF")
480 CALL CHAR(129,"818181FF81818181")
490 CALL CHAR(130,"01010101010204F8")
500 CALL CHAR(131,"808080808040201F")
510 CALL CHAR(132,"1F20408080808080")
```

```
520 CALL CHAR(138,"FBC4020101010101")
530 CALL CHAR(136,"000000FFFF")
540 CALL CHAR(137,"1818181818181818")
550 CALL CHAR(95,"000000007FFFF3F3F3F3FF7F00000000000000000
FFFF1F1F1F1FFFFF")
560 CALL CHAR(100,"00000000FF7FF8F8F8F87FFF00000000000000000
0FEFFCFCEFCFCFFFE")
570 CALL CHAR(104,"0B0F0F0F0F0C0C0C0F0F0C0C0C0F0F0F07D0F0F0F
0F0303030F0F03030F0F0E0")
580 CALL CHAR(108,"070F0F0F0C0C0F0F0C0C0C0F0F0F0F0BE0F0F0F
03030F0F0303030F0F0F0D0")
590 CALL CHAR(112,"014153333F3F1F1F1F1F3F7B6E8E0E01008183C
6EEFCFCF8F8F8FCFC9E0701")
600 CALL COLOR(13,16,2,14,16,1)
610 FOR I=2 TO 8 :: CALL COLOR(I,16,1) :: NEXT I
620 SUBEND
630 SUB INSTRUCTIONS
640 DISPLAY AT(10,8): "TRAFFIC COP."
650 DISPLAY AT(13,2): "DO YOU WANT INSTRUCTIONS?"
660 DISPLAY AT(15,11)BEEP: "(Y/N)"
670 CALL KEY(3,K,S):: IF S=0 THEN 670
680 IF K=78 THEN CALL CLEAR :: GOTO 760
690 IF K<>89 THEN 660
700 CALL CLEAR
710 DISPLAY AT(4,2): "USE THE ARROW KEYS (ESD,X)"
720 DISPLAY AT(6,3): "TO STOP THE TRAFFIC AND"
730 DISPLAY AT(8,1): "PREVENT COLLISIONS. WHEN TEN"
740 DISPLAY AT(10,3): "CRASHES HAVE OCCURED YOU"
750 DISPLAY AT(12,2): "WILL BE REMOVED FROM DUTY."
760 DISPLAY AT(13,4): "PRESS ANY KEY TO PLAY."
770 CALL KEY(3,K,S):: IF S=0 THEN 770
780 SUBEND
790 SUB SETUP(SF)
800 DISPLAY AT(4,1): "TRAFFIC"
810 DISPLAY AT(6,3): "COP"
820 DISPLAY AT(4,21): "SPEED"
830 DISPLAY AT(6,20): SP: "MPH"
840 DISPLAY AT(20,1): "TI-99/4A"
850 DISPLAY AT(20,21): "CRASHES"
860 CALL HCHAR(9,1,128,32)
870 CALL HCHAR(15,1,128,32)
880 CALL VCHAR(1,13,129,24)
890 CALL VCHAR(1,19,129,24)
900 CALL VCHAR(1,14,32,120)
910 CALL HCHAR(10,1,32,160)
920 CALL HCHAR(9,13,130)
930 CALL HCHAR(9,19,131)
940 CALL HCHAR(15,13,133)
950 CALL HCHAR(15,19,135)
```

```

880 FOR I=1 TO 22 STEP 3
890 CALL VCHAR(I,16,137,2)
900 NEXT I
910 SUBEND
920 SUB CARS(SP)
930 IF SP>20 THEN SP=SP-4
940 DISPLAY AT(6,20):SP;"MPH"
950 CALL SPRITE(#1,96,14,75,1,0,SP)
960 CALL SPRITE(#2,100,16,98,250,0,-SP)
970 CALL SPRITE(#3,103,6,1,173,SP,0)
980 CALL SPRITE(#4,104,3,1,1,105,-SP,0)
990 SUBEND
1000 SUB CRASH(CR)
1010 CALL MOTION(#1,0,0,#2,0,0,#3,0,0,#4,0,0)
1020 CALL SPRITE(#9,112,9,88,120)
1030 CALL SPRITE(#10,112,12,88,120,5,5,#11,112,12,88,120,-5,-5,#12,112,12,88,120,5,-5,#13,112,12,88,120,-5,5)
1040 FOR I=0 TO 20 :: CALL SOUND(-200,-7,I,110,I+5)
1050 CALL COLOR(#9,12) :: NEXT I
1060 CALL DELSPRITE(ALL) :: CR=CR+1
1070 DISPLAY AT(22,23):CR
1080 SUBEND
1090 SUB FASTER(CT,SP,BST)
1100 CALL SOUND(200,-2,0,500,5) :: SP=SP+2
1110 IF SP>BST THEN BST=SP
1120 CT=0 :: DISPLAY AT(6,20):SP;"MPH"
1130 SUBEND

```

