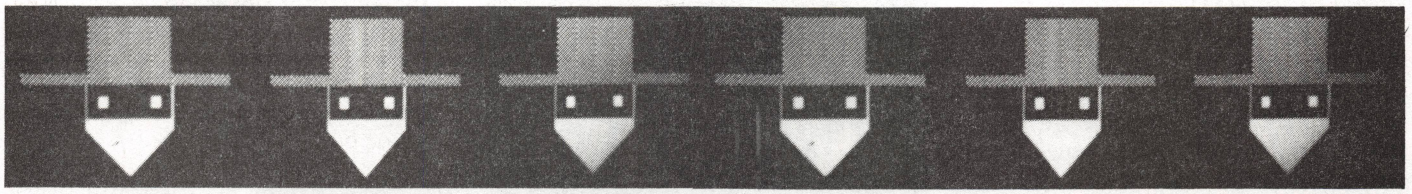
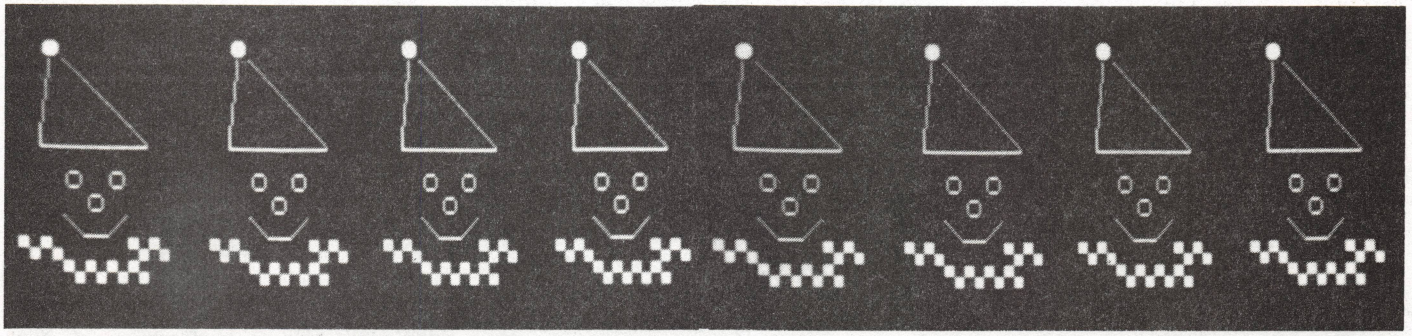


SPOT

The Society of PET Owners and Trainers



Commodore's PET is a factory assembled personal computer based on a 6502 microprocessor. The unit includes a keyboard, cassette tape unit, CRT, some graphics, upper and lower case, and an extended 8K BASIC. The system with 4K of user memory costs $595; the 8K model costs $795. Commodore has announced a 4 week delivery time for its $99.95 second cassette drive. Orders may be placed through Commodore, 901 California Ave, Palo Alto, CA 94304, (415) 326-4000.

PET fans will also be interested in the article on timing in this issue.

Photos in this article were taken of PET screens using a Polariod camera with a special hood to reduce glare.

We encourage readers to submit comments on and programs for the PET (or the TRS-80, or. . .). Please *submit only tested and debugged programs–we don't have time to debug programs here.*

SOFTWARE AVAILABILITY

Various ads for PET software have been appearing–we'd appreciate hearing from anyone who's tried any. Peninsula School software is offered under 'Software' in Announcements. We're still waiting for Commodore to begin selling software.

DOCUMENTATION

By now PET owners should have received their introductory manuals. Commodore will send, upon request, a collection of bulletins answering common questions PET owners have. A detailed PET User's Manual is now being written by Commodore; it will be sold for about $12–15. Data sheets on the chips used in the PET are available from KIM dealers.

PET NEWSLETTERS/GROUPS

In November, a professional-looking 7-page booklet bearing the title 'PET User Notes' was published by a PET enthusiast in Pennsylvania. Very general (but useful) information was provided. Future issues are scheduled to contain information on PET programs and systems offered for sale, and to devote much space to software exchange. The second of these bi-monthly publications should be out soon. To get on the mailing list, send $5.00 to PET User Notes, P O Box 371, Montgomeryville, PA 18936.

A PET Users' group is forming in Dallas to exchange ideas and information. Contact Carl Martin, 2001 Bryan Tower, Suite 3800, Dallas, TX 75201; (214) 742-5750.

Robert Elliott Purser (Box 466, El Dorado, CA 95623) is starting a reference list (to be published) of software available on cassettes for the PET, TRS-80, SOL and Apple II computers.

TAPE TIPS

At first we bemoaned the lack of a tape counter on the PET, as we suffered through hours spent searching for files on tape. But we've decided a tape counter wouldn't help that much, especially since we expect young children to use our PETs. Even with a tape counter you have to remember where on a tape a file is.

The solution we like best so far is to use short tapes–about 10 minutes long (5 minutes per side). That's enough tape to hold 2 copies of an 8K program on each side–but it's little enough tape that both sides of the tape can be used for program storage. (We gave up using more than one side of tapes longer than 15 minutes–it simply took too long to rewind.) We're pleased to announce we've come up with a source of inexpensive short tapes–see 'Cassette Data Tapes' under Other in Announcements.

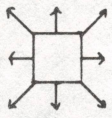
DRAW REVISITED

The PET drawing program has been expanded and modified by Larry Tesler and Dave Offen. We reproduce it here in full because portions of last issue's listing were pretty much unreadable.

How Draw Works. For new readers, we offer a brief summary of how DRAW works. For a more detailed explanation, see *People's Computers* Volume 6, Number 4; the program listed there will run on a 4K PET. The expanded version in this issue requires an 8K PET.

The program displays a dot as a drawing symbol at 'home', the center of the screen. The digit keys 1 through 9 are used to make the target move one cell in any of eight directions.

7	8	9
4	5	6
1	2	3

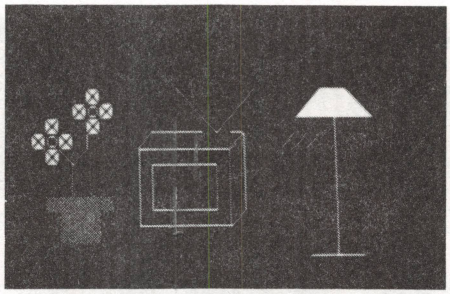


Pretend the target cell is on the '5' key. To move it left, press '4'; to move it up and right, press '9'; and so on. Whenever the target moves, it will inscribe the drawing symbol in its new cell. To change drawing symbols, press any graphic key; shift to get the graphics on keys 1-9.

RVS will reverse the drawing symbol. CLR and HOME work as usual. To erase, use SPACE. DEL erases the target cell without changing the drawing symbol. You can 'un-DEL' using the '5' key.

To move the target without changing the picture, get rid of the drawing symbol by pressing either of the CRSR keys, then use the digit keys to move the target. When you are ready to draw again, press a graphics key or RVS.

Dot Mode. The PET's graphics include small square dots which are each one-quarter size of a normal character. The 'dot mode' enables you to use the screen as a grid which can display up to 76 dots across and 48 down—twice as many symbols as the 'character' mode. To use the dots to draw, type shift->. The small square will become your drawing symbol and remain so until you press a graphic key, at which time you're back in 'character' mode. The new code used for dot



mode is in lines 4, 425, and 3100-3940; it is briefly described under 'Annotations.'

Repeating Keys. The number keys, when held down for a short time, will automatically start to repeat. For example, a line of symbols can be drawn across the screen by pressing the '6' key just once and holding it down for several seconds. The new code is in lines 25, 350, 2800–2850, and in the dot mode code. Memory location IB contains the number of characters in the keyboard input buffer. Memory location KD (discovered by Harry Saal for this purpose) contains 255 if no key is down and the row-column code of the down-key otherwise. It is possible that these locations may be different in future versions of the PET operating system.

Saving on Tape. The 8K version has been simplified to take advantage of the extra space. While the picture is being measured, a gray stripe creeps down the right side of the screen. While it is being stored into an array, the picture itself is covered with gray.

If, instead of a file name, an asterisk is typed (and a RETURN), then the picture will not be saved on tape. A BASIC program will be constructed that would print the picture left flush on the screen. That program is typed out with the word NEW above it and the cursor is put on the word NEW. If you wish to save the little printing program, press RETURN several times to erase the DRAW program and to enter each line of the printing program into memory. (You may want to change the line numbers first.) Finally, save the program the way any PET BASIC program is saved. Note: If the drawing is too wide, too high, or has too many reversals, then the printing program may not fit on the screen and none of this will work.

Lines 5510–5700 have been revised and lines 7000–7040 have been added.

Annotations for Dot Mode.

Q: quadrant number. For drawing small squares, each printing character is divided into four quadrants, numbered as shown:

1	4
2	8

QX: component of the quadrant in the X direction (0 or 1).

QY: component of the quadrant in the Y direction (0 or 1).

PT: pattern number. The pattern number is formed as the sum of all quadrant numbers required. For example, if quadrants 1 and 4 are filled, pattern 5 is used.

PN$ contains all characters comprised of small squares. Note that we print these special characters as words inside square brackets, since we have no way of printing graphic characters. Note: all characters we print inside square brackets are *shifted* (including SPACE) with the exceptions RVS, HOME, RIGHT (cursor right) and DOWN (cursor down). Lower case also requires shifting (as in line 30).

425 Enters dot mode when shift—> is typed.

3100-3140 clears screen, initializes variables, prints square in center of screen.

3150-3280 same as 100-900, for small squares.

3300-3330 same as 3000-3030, for small squares.

3400-3420 same as 4000-4020, for small squares.

3500-3520 same as 4500-4520, for small squares.

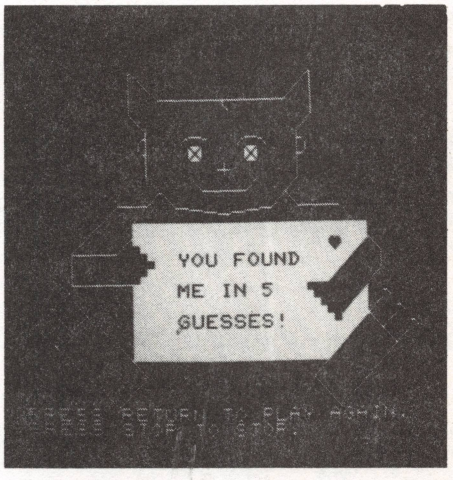
3600-3700 moves small square in direction indicated by number key.

3710-3780 reads current character from screen buffer so that other small squares within the same character will remain unchanged.

3790-3810 checks if target small square should be filled in (RV>0), blanked-out (RV<0) or left the same (RV=0).

3910-3915 replaces character on screen with pattern of small squares and blinks target square if necessary.

3920-3940 if number key is still down, moves again.



```
1 REM "DRAW 8K" FOR 8K PET
2 REM COPYRIGHT 1977, 1978
        PENINSULA SCHOOL, MENLO PARK, CA
3 REM PERMISSION TO USE, BUT NOT TO SELL
4 PNS="[OFF,SPACE,OFF,>,OFF,;,OFF,!,OFF,<,
   RVS,",RVS,?,RVS,COMMA,OFF,COMMA,
   OFF,?,OFF,",RVS,<,RVS,!,RVS,;,RVS,>,RVS,SPACE]"
5 LS="": I=0: POKE 59468,12
6 SC=32768: SP=32: SS=160: CV=191
7 BY=255: SH=128: US=127: QTS=CHR$(34)
8 LX=1: MX=38: LY=0: MY=24: DIM E$(MY)
9 HS="[LEFT]"+CHR$(0)+"[RIGHT]"
10 VS="[UP]"+CHR$(0)+"[DOWN]"
15 N1$="1": N9$="9": CR=13: RE=18: DE=20
20 WH=50: BL=53: WT=5: XR=164: IR=146
25 IB=525: KD=515
30 GR$="q": PRINT "[CLR]";
40 Y=INT((MY+LY)/2): X=INT((MX+LX)/2)
45 R=IR
50 PRINT "[HOME]";LEFT$("[12 DOWN]",Y);SPC(X);
55 IF PEEK(KD)<BY GOTO 55
60 GOTO 2300
100 GET CS: IF CS="" GOTO 3000
150 C=ASC(CS) AND US
200 IF FL>=WH THEN GOSUB 4000
250 IF C=CR THEN FL=-1E8: GOTO 100
300 FL=WH-WT
350 RP=0
400 IF CS>=N1$ AND CS<=N9$ THEN 1700
425 IF CS="[>]" GOTO 3130
450 IF C>=SP GOTO 1000
500 IF C=DE GOTO 4500
600 IF C=RE GOTO 1200
700 IF CS="[DOWN]" OR CS="[RIGHT]" THEN R=IR: GR$=""
750 IF CS="[HOME]" THEN GR$="": GOTO 40
800 IF CS="[CLR]" GOTO 30
850 IF CS="[LEFT]" GOTO 5000
875 IF CS="[UP]" GOTO 6000
900 GOTO 100
1000 GR$=CHR$(C+SH): R=IR
1100 GOTO 1300
1200 IF GR$="" THEN GR$=PG$: R=PR
1250 R=XR-R: PRINT CHR$(R);
1300 GOSUB 1400: GOTO 100
1400 PRINT GR$;"[LEFT]";
1500 PG$=GR$: PR=R
1600 RETURN
1700 PX=X: PY=Y
1750 X=X+C+1-3*INT((C+2)/3)
1800 IF X<LX THEN X=LX
1900 IF X>MX THEN X=MX
2000 Y=Y+1-INT((C-49)/3)
2100 IF Y<LY THEN Y=LY
2200 IF Y>MY  THEN Y=MY
2250 PRINT MID$(HS,X-PX+2,1);
2275 PRINT MID$(VS,Y-PY+2,1);
2300 L=SC+40*Y+X
2325 PG$=CHR$(PEEK(L) OR SH)
2350 PR=IR-(PEEK(L) AND SH)
2400 IF GR$<>"" THEN GOSUB 1400: GOTO 2800
2500 GOSUB 4000
2600 FOR DL=1 TO WT: NEXT DL
2700 GOSUB 4000
2800 IF PEEK(IB)<>0 OR PEEK(KD)=BY GOTO 100
2850 RP=RP+1: IF RP=1 THEN FOR DL=1 TO 200: NEXT DL:
                                    GOTO 2800
2900 GOTO 1700
3000 FL=FL+1
3010 IF FL=WH THEN GOSUB 4000
3020 IF FL=BL THEN FL=0: GOSUB 4000
3030 GOTO 100
3100 PRINT "[CLR]";
3110 Y=INT((MY+LY)/2): X=INT((MX+LX)/2)
3120 PRINT "[HOME]";LEFT$("[12 DOWN]",Y);SPC(X);
3130 RV=1: C=53: Q=1: QX=0: QY=0
3140 GOTO 3600
3150 GET CS: IF CS="" GOTO 3300
3160 C=ASC(CS) AND US
3170 IF FL>=WH THEN GOSUB 3400
3180 IF C=CR THEN FL=-1E8: GOTO 3150
3190 FL=WH-WT
3195 RP=0
3200 IF CS>=N1$ AND CS<=N9$ GOTO 3600
3210 IF C>=SP GOTO 425
3220 IF C=DE GOTO 3500
3223 IF C=RE THEN RV=(-RV) OR 1: GOTO 3800
3227 IF CS="[DOWN]" OR CS="[RIGHT]" THEN RV=0
3230 IF CS="[HOME]" GOTO 3110
3240 IF CS="[CLR]" GOTO 3100
3250 IF CS="[LEFT]" GOTO 5000
3260 IF CS="[UP]" GOTO 6000
3270 IF C>=SP GOTO 1000
3280 GOTO 3150
3300 FL=FL+1
3310 IF FL=WH THEN GOSUB 3400
3320 IF FL=BL THEN FL=0: GOSUB 3400
3330 GOTO 3150
3400 PT=PT+Q-2*((PT) AND Q)
3410 PRINT MID$(PNS,PT+PT+1,2);"[LEFT]";
3420 RETURN
3500 PT=(PT) OR Q
3510 GOSUB 3400
3520 GOTO 3150
3600 XC=C+1-3*INT((C+2)/3): YC=1-INT((C-49)/3)
3605 PX=X: PY=Y
3610 XT=X+X+QX+XC
3620 X=INT(XT/2): QX=(XT) AND 1
3630 IF X<LX THEN X=LX: QX=0
3640 IF X>MX THEN X=MX: QX=1
3650 YT=Y+Y+QY+YC
3660 Y=INT(YT/2): QY=(YT) AND 1
3670 IF Y<LY THEN Y=LY: QY=0
3680 IF Y>MY THEN Y=MY: QY=1
3690 Q=2↑(QX+QX+QY)
3700 PRINT MID$(HS,X-PX+2,1);MID$(VS,Y-PY+2,1);
3710 L=PEEK(SC+40*Y+X)
3720 LS=CHR$((L AND 63)+SH)
3725 PT=0
3730 FOR I=2 TO 16 STEP 2
3740 :IF MID$(PNS,I,1)=LS THEN PT=I/2-1: I=16
3750 NEXT I
3760 IF PT=0 GOTO 3800
3770 IF PT>=5 THEN PT=15-PT
3780 IF L AND SH THEN PT=15-PT
3790 IF RV=0 THEN GOSUB 3400: GOSUB 3400: GOTO 3920
3800 IF RV THEN PT=(PT) OR Q
3810 IF RV<0 THEN PT=PT-Q
3910 PRINT MID$(PNS,PT+PT+1,2);"[LEFT]";
3915 IF RV=0 THEN GOSUB 3400: GOSUB 3400
3920 IF C=RE OR PEEK(IB)<>0 OR PEEK(KD)=BY GOTO 3150
3930 RP=RP+1: IF RP=1 THEN FOR DL=1 TO 200: NEXT DL:
                                    GOTO 3920
3940 GOTO 3605
4000 PR=XR-PR: PRINT CHR$(PR);
4010 PRINT PG$;"[LEFT]";
4020 RETURN
4500 PG$=" ": PR=IR
4510 PRINT "[OFF] [LEFT]";CHR$(R);
4520 GOTO 100
5000 X0=MX: X1=LX: Y0=MY: Y1=LY: K=SC
5010 FOR Y=LY TO MY
5020 :FOR X=0 TO 39
5030 ::C=PEEK(K):K=K+1
5050 ::IF C=SP GOTO 5100
5060 ::IF X<X0 THEN X0=X
5070 ::IF X>X1 THEN X1=X
5080 ::IF Y<Y0 THEN Y0=Y
5090 ::IF Y>Y1 THEN Y1=Y
5100 :NEXT X
5110 :POKE K-1,102
5130 NEXT Y
5510 FOR Y=Y0 TO Y1
5512 :K=SC+40*Y+X0
5515 :RV=0: LS=""
5520 :FOR X=X0 TO X1
5525 ::C=(PEEK(K)-SP AND CV)+SS
5530 ::POKE K,102
5535 ::V=C>BY: IF V=RV GOTO 5545
5540 ::RV=V: LS=LS+MID$("[RVS, OFF]",V+2,1)
5545 ::K=K+1: LS=LS+CHR$(SH*RV+C)
5600 :NEXT X
5615 :E$(Y)=LS:
5620 NEXT Y
5630 INPUT "[CLR]SAVE FILE NAME OR *"; NM$
5635 IF NM$="*" GOTO 7000
5640 OPEN 1,1,1,NM$
5650 PRINT#1, Y1+1-Y0
5655 PRINT#1, X1+1-X0
5660 FOR Y=Y0 TO Y1
5665 :PRINT#1, QTS; E$(Y); QTS
5670 :E$(Y)=""
5675 :POKE 59411,53: T=TI
5676 :IF TI-T<3 GOTO 5676
5677 :POKE 59411,61
5678 NEXT Y
5690 CLOSE 1
5700 GOTO 30
6000 OPEN 1
6005 INPUT#1,Y1
6010 INPUT#1,X1
6015 IF ST GOTO 6100
6020 Y0=INT((MY+LY+1-Y1)/2)+1
6030 X0=INT((MX+LX+1-X1)/2)+1
6040 PRINT MID$("[CLR, 12 DOWN]",1,Y0);
6050 FOR Y=1 TO Y1
6060 :INPUT#1,LS
6070 :PRINT SPC(X0);"[LEFT]";LS;
6080 :IF Y<=MY THEN PRINT
6090 NEXT Y
6100 CLOSE 1
6110 GR$=""
6120 GOTO 40
7000 PRINT "[CLR, 2 DOWN]NEW"
7010 FOR Y=Y0 TO Y1
7020 :PRINT 901+Y-Y0;"?";QT$;E$(Y)
7030 NEXT Y
7040 PRINT"[HOME]";: END
```

PET PROJECTS & PROPOSALS

Lud Braun, developer of the Huntington Project simulations, has a PET. We're pleased that he sent us a copy of the letter he sent to Commodore so that we can share his reactions and activities with our readers.

I have had my PET for about a month now and thought that I should write down some of my observations about it. Basically, I am *very* pleased with it but, in typical human fashion, there are things I'd like to see improved. Among them are:

1. For educational use there should be a composite video signal coming out of a UHF or BNC Connector (not the odd-ball one that Radio Shack has used). This permits the teacher to bring the PET into the classroom to work with his entire class. I suggested this to you when I visited in early August and had hoped that it would have been added. We have designed and installed a mixer etc. and now have a composite video signal on old #77. It takes about $2 worth of parts.
2. The PET should have a handle on it. One of the first things I did after I got ours was to go to the hardware store to buy a 99¢ handle. It is very helpful when I carry the PET into class or home or onto an airplane. (You should see the airport security people freak out when I present this strange looking TV set for inspection!) I'm convinced that kids will be borrowing PETs over weekends from school or signing them out from the local public library, but they need a handle.
3. We have designed and installed a simple 4-channel A/D converter to permit us to enter parameter values into programs during execution of a simulation. This capability has enormous potential in education. It permits the digital computer to act like an analog computer (but with none of the problems of the real analog computer) and permits enormously improved simulation experiences. This also costs about $2 in parts.
4. We have had 6-10 system crashes in the month we've had it. We haven't been able to identify the cause although I suspect that it may be a voltage spike riding in on the line. The system suddenly hangs up and the keyboard goes dead. It had always been easy to recover just by turning the PET off and on, but then the memory is zeroed and any program I was devel-

oping has disappeared. Is it possible to restart without zeroing memory? Such a capability would sure be nice.

5. The monitor doesn't have equal gain horizontally and vertically. As a consequence, when I draw a circle (in a math program) it comes out an ellipse. A square (equal numbers of horizontal and vertical elements) comes out a rectangle. I have found a height adjustment in the monitor but no width control. Is there one? If so, I could fix it. (Incidentally, such controls probably should come out the back.)
6. It really would be nice to be able in one vertical and one horizontal command to send the cursor to any place on the screen (e.g., POKE 245, V). With this, graphing would really be easy. This amounts to a two-dimensional TAB command.

Despite these criticisms I am very happy with the PET. I am very excited about this machine for educational applications because of its powerful BASIC, because of its price, and because of its portability. The latter property is especially exciting because it gives us teachers, for the first time, the opportunity to let our students take a computer home overnight or over a weekend to develop an idea through use of a simulation or to solve a problem normally beyond their capability.

Ludwig Braun
Professor and Asst. Director
for Educational Technology
State University of New York
Stony Brook, NY

PET PICTURES

I was interested to read in the November-December issue of your experiences with the PET. Shortly after receiving the issue your DRAW program was running and we are now planning a contest for art students in one of the schools in which I am consulting. Using high contrast film and a micro lens, we are able to capture the work for reproduction. It is a most interesting exerise in applying the computer in some of the less expected areas of the school curriculum.

In addition to my own PET, we are getting a number of machines for the Simulation Center in the College of Osteopathic Medicine here at Michigan State University. Having followed John Starkweather's work at San Francisco, from the early days of Computest on the 1620 to his current efforts, I was most interested to read of your development of a PET version of PILOT. Is there more information available on that implementation? Is there any way this version can be purchased?

Norman T. Bell
Director, Faculty Development Program
College of Osteopathic Medicine
Michigan State University
East Lansing, MI

We'd enjoy seeing results of your art contest–some of our PET graphics accompany this article. As for your questions on PILOT, see the 'PET Software' ad under Announcements.

POST-DOC WARRANTY?

As a new PET owner, I found your November article timely and informative. I can't speak for the other owners but for one who has an elementary knowledge of BASIC and is interested in learning more BASIC plus assembly language, the PET to date has been a let-down. Foremost is their lack of a comprehensive instruction book which treats the three PET features: BASIC, assembly language and graphics. I argue that the lack of such a book does not allow me to fully test out the PET and therefore my warranty should start only after I receive my book. What do the other PET owners think?

The PET is a great machine, but there is always room for improvement. The case in point is the lack of a counter on the tape unit. Looking for programs on a cassette is a real chore. A counter would make PET even better.

I think the PET section in *People's Computers* is filling a real need, and I urge you to keep with it. We, the interested and uninformed PET owners of the world, support you.

Phillip Gash
Redding, CA 96001

A warranty that begins when you receive documentation is an idea that raises interesting questions, such as who gets to decide whether documentation is suitably 'comprehensive'? What if documentation is buggy–does this affect the warranty?

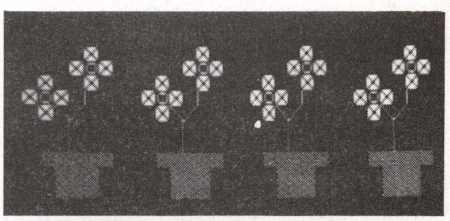
CHECKBOOK PROGRAM

```
70 REM ADD UP YOUR CHECKBOOK OR HOW TO
80 REM FORMAT FOR DOLLARS AND CENTS
90 REM BY OWEN HAWKINS
100 PRINT"[CLR]":N=1
120 INPUT "AMOUNT IS ";A
130 A=INT(A*100+.5): T=T+A
140 PRINT "[UP]";TAB(12);
150 PRINT "              ";: X$=STR$(A)
160 IF ABS(A)<10 THEN
      X$=LEFT$(X$,1)+"0"+RIGHT$(X$,1)
170 PRINT TAB(33-LEN(X$));"$";
180 PRINT LEFT$(X$,LEN(X$)-2);".";RIGHT$(X$,2)
190 IF A=0 THEN IF N THEN A=T: N=0: PRINT:
                    PRINT "TOTAL": GOTO 140
200 ON N GOTO 120
999 END
```

100 clears screen and sets GOTO flag N to 1.
120 rounds A to 2 decimal places and removes decimal point; computes cumulative total, T.
150 erases A on the screen and converts it to a string.
160 if A is less than 10 inserts a '0'.
170-180 prints the formatted number.
190 if finished (A was zero), then swaps T into A, sets flag to zero, and prints the total.
200 gets more numbers unless N=0.

For those of you who have not yet convinced your wife (and I suppose a husband or two) that buying the PET was a dire necessity, you might let her use the above program to add up the checkbook. It's considerably more fun than a pocket calculator and you can visually inspect your entries.

Unfortunately you are limited to numbers less than 10 million. If that presents a problem for you, I suggest that you simply do away with the cents by modifications to lines 140, 160 & 180. Now you can almost handle a billion. When you're ready for the total, enter '0' as the amount.

If you don't have a checkbook but like the formatted output, just use lines 140 to 180 in your own program. If you still don't like anything then at least take a closer look at line 190. The IF... THEN... THEN IF... etc can go fairly deep. And if you're big on commas (I'm not) the key is FROM 10 to 4 STEP –3.

Owen Hawkins
Menlo Park, CA