PET User Notes

Publication of the PET User Group P.O. Box 371 Montgomeryville, PA 18936

Volume 1 Issue 5 July-August 1978

Plot Routine
JS Barney Bryn Mawr, PA

Here's a simple plot routine to achieve maximum resolution from your graphics. It will be evident from an examination of your keyboard that two choices are available: maximum X resolution or maximum Y resolution.

Here's Y:

```
 10 D$="ddddddddddddddddddddddd":REM 23 CURSOR DOWN
 20 X$="rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrr":REM 39 CURSO
R RIGHT
 30 Y$="uuuuuuuuuuuuuuuuuuuuuuu":REM 23 CURSOR UP
 40 V$="        ":REM 8 HORIZONTAL BARS
 45 REM BARS ARE ON CHARACTERS $RF@CDE# ENTERED SEQUENTIALLY
 50 PRINT "c";D$;
 100 FOR X=1TO 28
 110 Y=(14-X)^2
 130 GOSUB 700
 135 NEXT X
 150 GET ZZ$:IF ZZ$=""GOTO150
 160 END
 700 XX=X:YX=Y/8:VX=(Y-8*YX)+1
 800 IF XX=0 GOTO 1000
 900 PRINTLEFT$(X$,XX);
 1000 IF YX=0 GOTO 1200
 1100 PRINTLEFT$(Y$,YX);
 1200 PRINTMID$(V$,VX,1);
 1300 PRINT"h";D$;
 1400 RETURN
READY.
c=clear,h=home,r=cursor right,d=down,u=up
```

Lines 10 through 50 establish strings in memory and initialize the screen.

Lines 700 through 1400 plot the point (X,Y) and return to main program.

Thus, the main program would contain lines 10 through 50 and subsequently a GOSUB 700 would be used to plot.

Please note that translation and scaling of the variables X,Y must assure that $0 \leq X\% \leq 40$ and $0 \leq Y\% \leq 23$.

If you are unsure of the scaling and translation, then enter conditional statements to exit and/or flag to prevent scrolling the plot.

****

```
1 REM ROY BUSDIECKER
2 REM TO FIND THE REMAINDER (R) IN INTEGER DIVISION
3 REM    (Q IS THE QUOTIENT):
4 REM
10 INPUT "DIVISOR";M
20 INPUT "DIVIDEND";N
25 IF M=0 GOTO 10
30 Q=INT(N/M)
40 R=N-M*INT(N/M)
50 PRINT"QUOTIENT";Q;"REMAINDER";R
60 GOTO 10
```

GENERAL

Commodore raised the price of their printer from $595 to $695. Although the increase is significant, the price/performance is still very attractive (I ordered my printer before the increase, I hope). Deliveries are scheduled for October.

At the Chicago Consumer Electronics Show, Commodore exhibited a dual drive increased density minifloppy system (260,000 bytes?). The anticipated price is $1000, with delivery in early 1979.

None of the Commodore announced software has been shipped yet. Neither has Commodore sent documentation or User Manuals. Other than the 019 ROM replacement to fix the cursor loss, none of the significant system and BASIC bugs have been corrected. Several people have also written they had trouble even acquiring the cursor fix ROM.

Commodore also announced a PET schematic-diagnostic package for sale at $30. Included will be: edge connectors (for running diagnostics); diagnostic routine cassette; diagnostic package manual; and schematics of the circuit boards. Delivery is 4 weeks (no comments like "from when?" allowed).

Commodore plans to have ROM fixes for many of the BASIC and operating system bugs in about 10 weeks, along with Machine Language Monitor in ROM.

ANNOUNCEMENTS

CGRS Microtech, Box 368, Southampton, PA 18966, (215) 757-0284 announced PETEX, a PET to S-100 single board adapter card which will plug into a slot on an S-100 system. The board generates all S-100 bus signals, and comes assembled and tested for $195. CGRS also announced disk software for 6502 systems to support the Versafloppy disk controller. Supplied on a 2708 EPROM and diskette at price of $40.

ML Sandy, Box 1535, Corinth, MS 38834 has coding-documentation pads of about 50 sheets for PET at $2 or 5 for $7.50 postpaid. Each page has all PET character representations as well as a 40*25 screen layout on the back.

Creative Computing (Creative Computing Software, Box 769-N, Norristown, NJ 07960) is looking for PET software to market on a royalty basis -- generally 10-15% of a retail price of $7.95. Send software on cassette along with any necessary documentation and return postage.

****

PET User Notes published 6 (or more) times a year by Gene Beals, PET User Group, Box 371, Montgomeryville, PA 18936. Membership/subscription on annual basis is $5 (U.S.) in U.S. and Canada, and $10 in other countries.

Connecticut microComputor, 150 Pocono Road, Brookfield, CT 06804 has a word processing program for PET for $29.95. The software supports shift reversal, upper and lower case, tape storage, output to printer, and all the necessary word processing/editting commands. (The CmC Word Processor is being used for this section and seems to work quite well.)

Microsignal, Box 161988, Sacramento, CA 95816 has announced COMPUVOX, a low cost ($29.95) voice recognition unit for PET. Compuvox apparently recognizes number and duration of sounds (not the actual word). Two programs (Voicetrap and Voicemaze) are available for $3.95 each to demonstrate Compuvox. Sounds like fun.

Microtronics, 5943 Pioneer Road, Hughson, CA 95326, (209) 634-8888 announced the M-65 Ham Interface for PET. The package includes the hardware, and programs MORSE and RTTY on cassette for $99.95. The system was designed by Ron Lodewyck (N6EE) one of the co-authors of Commodore's Basic BASIC (forthcoming?).

On a related subject, Larry Williams (N5CX), San Antonio , mentioned a group of ham PET owners meet each Sunday morning at 15:00 UCT (10:00AM CDT) on 14240 KHz.

Stefan Ram, Wedellstr. 23, 1000 Berlin 46, GERMANY has some programs on cassette to sell ($10 for entire cassette incl. air mail) or trade for equivalent tapes. His tape includes: a basic file system with up to 10 record elements; a descriptive program; and several miscellaneous items including a music program and several math plotting routines.

I received a series of five workbooks from Total Information Services, Box 921, Los Alamos, NM 87544. They are entitled: Getting Started with Your PET; Pet String and Array Handling; PET Graphics; PET Cassette; and PET Miscellaneous. The books appeared to be nicely done. While they start out at a beginning level, the author still assumes you have had some exposure to BASIC (such as one of the many available BASIC textbooks).

Program Design, Inc., 11 Idar Court, Greenwich, CT 06830 has several educational programs for PET, including vocabulary builder and number practice.

Russell Grockett, 401 Monument Rd, #177, Jacksonville, FL 32211 is part of a group offering low cost PET software ($2.50 for first program and tape, $1.50 each additional program). Write for list of programs.

Edward Lichten, 10301 Falling Tree Way, Louisville, KY 40223 has programs for exchange (STATECAPS,MATH by grade level, Biorythm, letter recognition for preschoolers).

Rick Goldsmith, 2730 Townway Rd. No. E54, Danville, IL 61832 is interested in the possibility of a user group in central Illinois. Contact him if you are in that area.

As we were ready to take this issue to the printer, I received a copy of PET FORTH (a version of their 6502 FORTH) from Programma Consultants, 3400 Wilshire Blvd, Los Angeles, CA 90010. Programma stated they would have a Primer and full User Reference available in the near future. Hope to have more on this in the next issue.

MISCELLANEOUS

Several people (Grant Paul, Charley Musselman among others) wrote that the random number can be almost randomly seeded after PET has been cold-started by using the negative value of the jiffy check: X=RND(-TI)

Users in the California Bay area reported seeing a demonstration of a very nice PET assembler, link editor, and text editor all written in machine language. The package will be available in the near future.

A number of people also suggested clear nail polish as a means of protecting keytops. A slightly more elegant sounding suggestion from Fred Mellender was Grumbacher Hyplar Gloss Medium from art supply stores. Frank Covitz suggested Dennison's Pres-A-Ply Clear Seal plastic sheet.

I have signed up for one of the user group meeting rooms at PC 78. The registration was late, however, so I don't know if we'll get the room. If not, I'll be at the ACG-NJ 6502 group meeting area.

Have received good comments from people about the following units:

BETSI, Forethought Products, Box 8066, Coburg, OR
EXPANDAPET, Convenience Living Systems, 648 Sheraton Dr., Sunnyvale, CA 94087
PEM-8K, International Technical Systems, Box 264, Woodbridge, VA 22194

Saw some of the CMS Casino series programs (Blackjack, Baccarat, etc.), and thought they were very nicely done -- good graphics and good descriptions of both the actual games and program operation.

We have about 4 pages of memory map and memory locations. Since I did not feel everyone is interested in these, I didn't include them in this issue. If you want a copy, please send stamped self addressed envelope and 2 extra stamps (unattached) for copy costs.

Received "24 Second Qubic" program from Mike Louder. Although it plays much faster than the Kilobaud original, the primary intent was to demonstrate program run time enhancement by dynamically modifying BASIC code. Mike will hopefully have an article on this by next issue.

I recently purchased a printer with RS-232 interface since Commodore is not yet shipping theirs (printer). To get up and running quickly, I obtained an IEEE to RS-232 adapter from Connecticut microComputer. The interface worked without a hitch, and most of the listings you see in this issue were done on the printer and adapter.

Arthur Hudson, Ottawa, Ontario suggested we run a contest to find the program which does the most in 4 lines. His entry:

```
10 PRINT"clr screen":FORI=1TO999
20 PRINTCHR$(INT(2*RND(2))+205);:NEXT
30 POKE33307,83
40 FORJ=1TO12000:NEXTJ:GOTO10
```

The program draws a maze and allows about 16 seconds to get out.
Mr. Hudson also suggested a printer as the prize, and further inferred that I should send him the printer right away since no one would best his entry. While I think the contest is OK, I would like to revise the prize structure : an autographed listing of Mr.Hudson's program to anyone surpassing the above entry. For Arthur Hudson a certificate of merit printed on the back of used Mod 33 teletype paper. A clever program anyway -- thanks Arthur.

Thanks also to Neil Bussey of SPHINX for sending the programs Resequence (by Joe Trimble) and Unlist (by Larry Tessler) which are listed in this issue, and for HIMONDIS (monitor-dissasembler).

Errors in V 1,Issue 4

Andrew Fraley, 1753 York Road, Reading, PA 19610 is selling Bomber, Indy, Seawolf, and Dogfight for $5.95 (for all four, not each).
Page 2 -- "Sensing Keys". Shift depression checked in location 516.
Page 3 Cosine abbreviation should be COs (who would do it, though).
Page 6 & 7 -- "LIFE" 19DF should contain 60, and 1A68 should have D0 CF.

Programs on Tape

SEQUENCE -- you have to sort a character list
STATECAPS -- quiz on states and capitals from Kilobaud 3/78
RACETRACK -- see listing in V.1, Issue 3 - RE McGee
DEFLECTION -- F Dunlap
ADDITION -- math drill -- F Dunlap, C Pitcairn
OTHELLO -- from Byte 9/77
OTHELLO for 2 -- F dunlap
BAGELS -- like Mastermind - J Butterfield
Cash flow/ Return on Investment - R Goldsmith
STAR TREK
BLACKJACK -- D Liem - nice graphics, humorous patter by dealer
LIFE -- 40*25 F Covitz
LIFE -- 64*64 F Covitz
TRAP -- trap the PET, avoid being trapped
SUPER MASTERMIND -- JR Marcou
LUNAR LANDER -- JR Marcou
NIM -- J butterfield
QUBIC -- 3D Tic-Tac-Toe
LIST MEMORY -- Charles Combs
Machine Language Monitor -- a good version - similar to the one Commodore will be sending
TIME -- large clock display
BIORYTHM -- Kenneth Finn - very concise code - plots with only 30 BASIC statements
KING -- business-social simulation or game
BREAKOUT -- simple version of the arcade game
SWATPLOT -- F Campbell (see V.1 Issue 3) - plots in 80x50 format
Word Processor -- M Richter, R Julin - no printer interface yet
MAXIT -- Harry Saal - clever game and nicely programmed
WUMPUS
MARKET -- corporate simulation
CONCENTRATION -- Francis Chambers - displays 52 cards
ESP TEST -- Francis Chambers
DO-ALL -- JK Johnson from Kilobaud
SHARK BAIT -- JK Johnson - hangman stye program
FLEA RACE -- JK Johnson
AWARI -- Hans-J Koch - German instructions
CHASE with Sound
STAR LANES -- Gerald Hasty - Interface Age
KALEIDOSCOPE -- Jerry Panofsky - People's Computers
HEXDEC -- Wayne Reindollar -- converts and pokes values into memory for machine language programs
24 Second QUBIC -- Mike Louder
DOODLER -- Jim Brannan - draw type program
GNIP GNOP -- Jim Brannan - 2 player ping pong
PONG -- People's Computers
CURFIT -- J Butterfield - fits data to 6 curves
TRIANGLE -- J Butterfield - solves any triangle
METRIC -- J Butterfield - does metric converstions
DATES -- J Butterfield - day of week, days between
TRENDLINE -- J Butterfield - fits, forecasts, graphs
MILEAGE -- J Butterfield - distances from Latitude/Longitude
FACTORS -- J Butterfield - prime factors of any number
MORTGAGE -- J Butterfield - schedule of payments
FINANCE -- J Butterfield - present, future value etc.
BATTLESHIPS -- J Butterfield - you vs. computer
MOONLANDER -- J Butterfield - graphics
BREAKOUT -- J Butterfield - paddle ball
CRYPTO -- J Butterfield - cryptogram solving aid
JOTTO -- J Butterfield - guess a word
POEMS -- J Butterfield - write poetry
MYSTERY -- J Butterfield - you can't list it!
ADDER -- Earl Wuchter - functions as an adding machine (nicely done)
HIMONDISS -- monitor and disassembler from SPHINX
RESEQUENCER -- Joe Trimble - renumbers BASIC programs by 10
UNLIST -- Larry Tesler - in this issue
ELIZA -- adapted for PET by Dennis Cumberton - the computer psychologist
HAMMURABI -- social simulation
SLOT MACHINE -- Michael Richter
CRAPS --Michael Richter
BREAKOUT with Sound
RESEQUENCER -- Jim Trimble - line renumber routine
POP SHOT -- from Sphinx Group - shooting gallery with sound
STARS -- John Broomhall - children's number guessing game
LINEQN--Frank Alexander - solves linear equations using matrix invert subroutine.
POUNCE--John Broomhall - kids game--if you don't pounce the right number of spaces, the mouse might run into his hole.
STAR WARS--John Broomhall
AUTO-DOODLE--Frank Levinson - draws very nice rectangular patterns.
FOURIER--Frank Levinson - very nice high density graphing--visual demo of Fourier approximation curve fit.

$2 for the first program including cassette and postage, $1 tor each additional -- max 4 per tape. This "programs on tape service" is priced to hopefully cover the cost of a part time person to copy and mail the tapes. If you have a program to add to the list, please send it on tape. We will copy it and return your tape (with program of your choice).

PET Operator Hierarchy
Raynor Taylor Charleston, South Carolina

I have been a witness to a phenomenon of the PET that is as yet unmentioned. I am very excited about this phenomenon particularly because it is usually found only on the large computers. Sometimes when I try to execute a machine language subroutine, the PET gives me an "ILLEGAL QUANTITY ERROR". After I recheck the argument in SYS(X) or USR(X) and find it to be within the prescribed ranges, the PET still refuses to execute the subroutine. At this point I figure that something must be wrong with the subroutine, so I go back and check it. What do you know, I always find an error.

What this means is that the PET, before giving itself up in a possibly detrimental subroutine, looks over and checks for errors. Some type of dynamic debugger must be implemented in the PET - imagine, a program that checks programs!.

Sometimes, however, the dynamic debugger will fail to find logical errors such as endless loops, etc. This is because the program looks good, that is, has no illegal opcodes, no bad branches etc.

```
'RESEQUENCER' BY JOE TRIMBLE
  OAKLAND, CA.  5/29/78

THIS PROGRAM WILL RESEQUENCE LINE NUMBERS (INCLUDING ALL GO
TO'S, GOSUB'S AND THEN'S) BY 10, STARTING WITH 10, FOR ANY
SUBSEQUENTLY LOADED PROGRAM.
  THIS PROGRAM HAS SEVERAL RESTRICTIONS:
1. LINE NUMBER 0 WILL BE RESEQUENCED TO LINE 10, BUT
  REFERENCES TO LINE 0 WILL NOT BE CONVERTED.
2. LINE NUMBERS GREATER THAN 63987 WILL NOT BE RESEQUENCED
3. ONLY 255 LINES CAN BE CORRECTLY RESEQUENCED, DUE TO
  THE CURRENT MATRIX LIMIT.
4. ALL NUMBERS FOLLOWING ANY 'GOTO', 'GOSUB', OR 'THEN'
ON THE SAME TEXT LINE WILL BE CONVERTED, IF POSSIBLE!
(EVEN AFTER A ':' OR INSIDE PARENTHESES.)
5. IF THE NEW (RESEQUENCED) NUMBER HAS MORE DIGITS THAN
THE OLD NUMBER, THE CHARACTER  PRECEDING THE OLD NUMBER IN
TEXT LINES IS REPLACED BY THE FIRST DIGIT OF THE NEW NUMBER
FOR EXAMPLE, 'GOTO 25' BECOMES 'GOTO150' (IF 150 IS
  THE NEW EQUIVALENT OF OLD LINE 25).
   HOWEVER, IF NO SPACE PRECEDES THE OLD LINE NUMBER,
THE 'GOTO', 'GOSUB', 'THEN' OR COMMA (IN THE CASE OF
  'ON...THEN ##,##,##') WILL BE REPLACED BY THE FIRST
  DIGIT OF THE NEW NUMBER.
   FOR EXAMPLE, 'GOTO25' BECOMES JUST '150' (IF 150
IS THE NEW EQUIVALENT OF OLD LINE 25).  'ON...THEN
  25,26,27' BECOMES 'ON...THEN150160170'

TO RUN THIS PROGRAM:
1. LIST LINES 63988-63999
2. REPLACE 'READY' AFTER  LISTING WITH 'LOAD'-DON'T 'RETURN'
3. PRESS 'PLAY', THEN PRESS 'RETURN'.
4. AFTER 'READY' APPEARS, GO 'HOME', FOLLOWED BY 11 'RETURN'S
TO ENTER LINES 63988-63999 INTO YOUR PROGRAM.
5. TYPE 'GOTO 63988', THEN 'RETURN'.
6. AFTER 'READY' APPEARS SAVE THE RESEQUENCED PROGRAM.
(YOU MAY DELETE LAST 11 LINES FIRST UNLESS NEEDED AGAIN)

 63988 DIML(256):L=1025:DEFFNR(X)=PEEK(X)+256*PEEK(X+1):DEFFNM
(X)=INT(10*X/256)
 63989 N=FNR(L):X=FNR(L+2):IF X<63988 THEN A=A+1:L(A)=X:L=N:GO
TO 63989
 63990 L=1025:FOR B=1 TO A:N=FNR(L):POKE(L+3),FNM(B):POKE(L+2)
,10*B-256*FNM(B)
 63991 F=0:FOR C=L+4TON-1:P=PEEK(C):IF P=137OR P=141OR P=167TH
EN F=1:GOTO 63999
 63993 IF F<>0 THEN IF P>47 AND P<59 THEN D=10*D+P-48:G=G+1:GO
TO 63999
 63994 IF F=0 OR D=0 GOTO 63999
 63995 FOR E=1 TO A:IF D=L(E)GOTO 63997
 63996 NEXT E:D=0:G=0:GOTO 63999
 63997 D=0:E$=STR$(E*10)+"    ":H=LEN(E$)-4:C=C-G:IF H>G THEN C
=C-1:G=H
 63998 FOR I=1 TO G:POKE C,ASC(MID$(E$,I+1,1)):C=C+1:NEXT I:G=0
 63999 NEXT C:L=N:PRINT B;:NEXT B:END
```

REPEAT Key
Hans-J. Koch West Germany

The sensing of keys with location 515 is very good (issue 4), when you have a program for text editing. One thing that the PET lacks is a REPEAT, or automatic, key. With the following short program, you can have a repeat function:

```
10  GET A$: IF A$="" GOTO 10
20  P=PEEK(515): D=60
30  PRINT A$;
40  T=TI+D
50  IF PEEK(515)=255 GOTO 10
60  IF TI<T GOTO 50
70  D=6: GOTO 30
```

The program echoes the input. If you release the key, the program waits for the next input. If you press the key more than one second (D=60), you will get the character ten times a second (D=6). If you press a second key, then release the first key, the program will not work properly, but that can be corrected with a few modifictions.

KIM-1 Talks to PET

Frank Covitz

Since both the KIM-1 and the PET are 6502 based systems, it ought to be possible to convert many KIM programs to run on the PET. Here are a couple of programs to enable the KIM-1 to transfer data to the PET over the user ports, with KIM providing a "handshake" signal to PET.

Both programs are relocatable, except for the JMP address at the end of each, which sets up things for a repeat. Stopping at the BRK is OK. (The KIM appears to jump over 2 bytes after a BRK, but the PET seems to jump over only one byte). Don't forget to set the BRK address on the KIM, at address 17FE and 17FF, to 00,1C .

I have a copy of a PET machine language monitor, which makes it fairly easy to program the PET. By the time this article is printed, the PET monitor should be available (according to Commodore). Without the monitor the PET data has to be POKE'd in decimal.

Both programs should be fairly clear to KIM programmers. The KIM fetches data bytes from CURL,H and puts them out to the port A lines. A handshake signal (high to low to high) is then sent over KIM port B-0 to the PET CA1 line. This signals that a new byte is available. The JSR SCAND at 17AE was put in to let the KIM LED display show the data flow, and limits the transfer speed to about 1 page (256 bytes) per second, fast enough for me. By eliminating the JSR SCAND (with NOP's), and setting the delay time byte at 17AA to 01, I have been able to get reliable data transfer speeds of about 30,000 bytes per second!! (The SEI and CLI instructions in the PET programs are required to get that speed).

The PET program is fairly straightforward. On turn-on the CA1 port is set to recognize a high-low-high transition (this clears bit 1 of the IFR control register at $E84D_{hex}$, which resets itself when the user port is read -- called SYNC in the PET application notes and is at address E841). Note - the zero page addresses 00 through 05 are used in the PET program and may not be "safe" on return to BASIC. Therefore, without the monitor, it may be necessary to restore them before the BRK. On my machine these six bytes are 76,48,209,0,0,0 (decimal) or 4C,30,D1,00,00,00 (hex).

As mentioned above, connect the KIM port A0-7 lines to the corresponding PET user port lines (PA0-7, pins C-1 on the user port connector), and the KIM port B0 line to the PET CA1 line (pin B on the user port). Set the start address (low,high), and the end address + 1 (low,high) of the KIM data to be dumped into address 0000-0003, respectively. Do the same for the PET addresses 0000-0003 (to where the data will be dumped). The PET addresses do not have to correspond (except in that the number of bytes to be dumped should be equal or less than the KIM data to be transferred.

Start the KIM program first (at address 1780_{hex}) then start the PET program at 0800_{hex} (or 2048_{dec}). Nothing happens until the "+" key on the KIM is pressed. If you have left in the JSR SCAND instruction on the KIM, you will see the addresses and data being transferred on the KIM LED display.

One further note on transcribing KIM programs to run on the PET. Be careful to convert all port addresses and data direction registers, all references to absolute addresses. References to KIM ROM routines will, of course, need to be re-written.

KIM-1 Program (part of KIM Talks to PET)

```
KIM Address Op.code   Byte 1,2    Addr. Op. B1 B2    Comments
    START   LDAim     FF          1780  A9  FF       Set up Port A0-7 as
            STA       PADD        1782  8D  01 17    outputs
            LDAim     01          1785  A9  01       Set up Port B0 as
            STA       PBDD        1787  8D  03 17    ouput and set the
            STA       PBD         178A  8D  02 17    bit to 1
    BACK    JSR       GETKEY      178D  20  6A 1F    Wait for the "+" key
            CMPim     "+"         1790  C9  12
            BNE       BACK        1792  D0  F9
            LDAz      STARTL      1794  A5  00       Set up start address
            STAz      POINTL      1796  85  FA
            LDAz      STARTH      1798  A5  01
            STAz      POINTH      179A  85  FB
    AGAIN   LDYim     00          179C  A0  00       Straight indirect
            LDAiy     POINTL      179E  B1  FA       Fetch byte
            STA       PAD         17A0  8D  00 17    Put it on Port A
            DEC       PBD         17A3  CE  02 17    Put out a low-to-high
            INC       PBD         17A6  EE  02 17    transition on Port B0
            LDAim     Delay       17A9  A9  10       Slow down
            STA       TIMER       17AB  8D  04 17
    DISPL   JSR       SCAND       17AE  20  19 1F    Show it on display
            BIT       TIMER*      17B1  2C  07 17    Time up?
            BPL       DISPL       17B4  10  F8
            INCz      POINTL      17B6  E6  FA       Now get address of next
            BNE       COMP        17B8  D0  02       byte
            INCz      POINTH      17BA  E6  FB
    COMP    LDAz      POINTL      17BC  A5  FA       Check for end
            CMPz      ENDL        17BE  C5  02
            BNE       AGAIN       17C0  D0  DA
            LDAz      POINTH      17C2  A5  FB
            CMPz      ENDH        17C4  C5  03
            BNE       AGAIN       17C6  D0  D4
            BRK   BRK  .          17C8  00  00       Done
            JMP       START       17CA  4C  80 17    Back to beginning
```

```
MNEMONIC  ADDRESS
STARTL    0000          This program is relocatable, except
STARTH    0001        for the JMP  START at  17CA
ENDL      0002          For high speed replace JSR  SCAND (20 191F)
ENDH      0003        at 17AE with NOP NOP NOP (EA EA EA).
POINTL    00FA
POINTH    00FB
PAD       1700
PADD      1701
PBD       1702
PBDD      1703
TIMER     1704
TIMER*    1707  (the time-out addr.)
START     1780
BACK      178D
AGAIN     179C
DISPL     17AE
COMP      17BC
SCAND     1F19
GETKEY    1F6A
```

P.E.T. Program (part of KIM talks to PET)

```
PET Address Op.code   Byte 1,2    Addr. Op. B1 B2    Comments
    BEGIN   SEI                   0800  78           Disable interrupts
            LDA       IFR         0801  AD  4D E8    Set up IFR to recognize
            ORAim     02          0804  09  02       a high-to-low transition
            STA       IFR         0806  8D  4D E8
            LDAz      STARTL      0809  A5  00       Set up start address
            STAz      CURL        080B  85  04
            LDAz      STARTH      080D  A5  01
            STAz      CURH        080F  85  05
            LDYim     00          0811  A0  00       Straight indirect
    AGAIN   LDAim     02          0813  A9  02       Check CA1 for strobe
    BACK    BIT       IFR         0815  2C  4D EF
            BEQ       BACK        0818  F0  FB
            LDA       SYNC        081A  AD  41 E8    Fetch data from user port
            STAiy     CURL        081D  91  04       Store it
            INCz      CURL        081F  E6  05       Now get next address
            BNE       COMP        0821  D0  02
            INCz      CURH        0823  E6  05
    COMP    LDAz      CURL        0825  A5  04       Check for end
            CMPz      ENDL        0827  C5  02
            BNE       AGAIN       0829  D0  E8
            LDAz      CURH        082B  A5  05
            CMPz      ENDH        082D  C5  03
            BNE       AGAIN       082F  D0  E2
            CLI                   0831  58           Clear interrupt disable
            BRK                   0832  00           Stop
            JMP       BEGIN       0833  4C  00 08    Get set for repeat
```

```
MNEMONIC  ADDRESS
STARTL    0000          The program is relocatable, except
STARTH    0001        for the JMP BEGIN at 0833
ENDL      0002
ENDH      0003
CURL      0004
CURH      0005
BEGIN     0800
AGAIN     0813
BACK      0815
COMP      0825
SYNC      E841 (user port with handshake
               pre-set as all input on turn-on
               of PET)
IFR       E84D (Control register for user port)
```

Reversing Upper/Lower Case
Jim Butterfield Toronto

Some people seem to be bothered that if you perform POKE 59468,14 to permit lower case characters, the keyboard is reversed - that is, you have to use SHIFT for lower case letters. It's easy enough to fix, if desired, as the simple demonstration program shows:

```
300  POKE 59468,14 : C=32        (space character)
310  C=198-C : T=TI+15 : ? CHR$(C) ;"cl";     ("cursor left")
320  GET Z$ : IF Z$<> "" GOTO 350
330  IF TI < T GOTO 320
340  GOTO 310
350  Z=ASC(Z$) : IF Z > 64 AND Z < 91 THEN Z$=CHR$(Z+128)
360  IF Z > 192 and Z < 219 THEN Z$=CHR$(Z-128)
370  ? "sp cl";Z$; : GOTO 320          ("space cursor left")
```

Special PET Print Control
Jim Butterfield Toronto

By now, most readers of the instruction book - and PET User Notes know about the cursor movements, plus Home and Screen Clear. A couple if others should be mentioned:

-- Delete (reverse T) can be useful because it moves everything to the right of the cursor (on the same line) to the left. Great for generating left-movement effects. Remember that Delete doesn't work if the cursor is in column 1.
-- Insert (reverse shift T) gives the opposite effect to Delete: it will move a whole line right. Remember that you must follow the Insert symbol with the information to be inserted (say, a space); and characters pushed beyond the right margin may reappear on the next line.
-- Carriage Return (reverse shift M) starts a new line.
-- Reverse (reverse R) and reverse-off (reverse shift R) turn the reverse feature on and off during printing.

SWITCHER
Frank Covitz Lebanon, NJ

Since most 6502 programs make extensive use of zero-page, here is my SWITCHER routine:

This is locatable anywhere. To use, stick a JSR SWITCHER at beginning of your 6502 program then do another JSR SWITCHER before the final BRK, RTS, or JMP BASIC.

```
SEI                  78        protect from interupts
LDXim #BYTES         A2 NN     NN=no. of bytes to be preserved
LDAzx BASE-1         B5 MM     MM=start addrlow-1 of zero-page
PHA                  48        save it on stack
LDAx BASE-1 PAGEH    BD MM PP  fetch data from page PP
STAzx BASE-1         95 MM     store it in page zero
PLA                  68        fetch previous page zero data
STAx BASE-1 PAGEH    9D MM PP  store it in a safe place-SWITCH
DEX                  CA        do it NN times
BNE F1               D0 F1
CLI                 58        clear the interrupt disable bit
RTS                 60        then return
```

This is locatable anywhere. To use, stick a JSR SWITCHER at beginning of your 6502 program then do another JSR SWITCHER before the final BRK, RTS, or JMP BASIC.

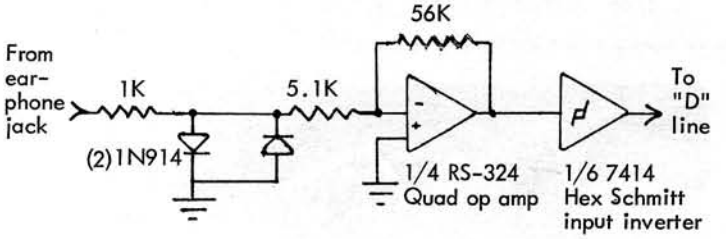
****

```
40 REM  WOLF
50 REM NEIL HARRIS
70 REM CB2 ON USER PORT TO AMPLIFIER
100 POKE59467,16:POKE59466,15
110 FORL=180TO76STEP-3:POKE59464,L:NEXT
120 FORL=200TO100STEP-3:POKE59464,L:NEXT
130 FORL=100TO250STEP3:POKE59464,L:NEXT
999 POKE59467,0
```

Some Possible Modifications to Second Cassette Interface from Volume 1, Issue 3

Stephen Hui, using a Sony recorder, needed as extra inverter on the input circuit (which he obtained by tying in another gate on the same 74132 package).

Frank Covitz sent the following:



Apparently Frank's Sear's cassette recorder has either low output and/or high distortion.

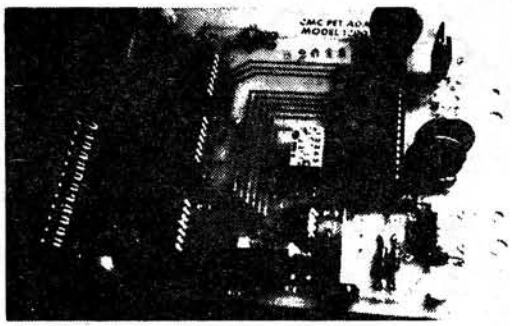
The diodes and ~10X DC amplifier work over a wide range of input voltages. Both IC's are driven from the PET 5V "B" line. I have had very good reliability on both read and write, and tape interchangeability using this cheap recorder, which, by the way, has a tape counter. The tape itself seems to be the most critical element. I have had very reliable results using Radio Shack Super-tape and Ampex Plus Series cassette tape.

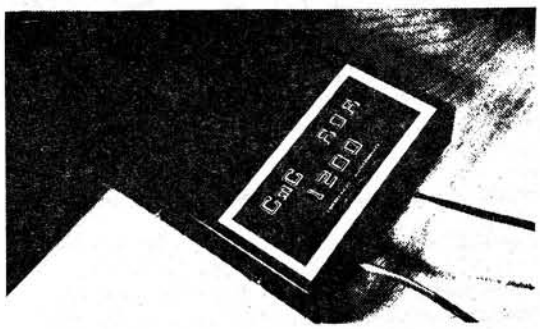
Russell Martin suggested that the problem is probably with the recorder, and that any player used should be capable of producing TTL switching level voltages at the inputs of the Schmidt trigger NAND gate. His Bigston KD-300Q has an output of 4.8VAC RMS and works well, but his Craig 2603 produces 3.8VAC RMS, and isn't recognized by the interface (due to additional voltage drop caused by the 1N914 diode and the half wave rectifier circuit on which it was used). Possible solutions: replace 1N914 (or 1N4148) with a germanium diode; use full wave rectifier bridge to replace single diode; use IC audio amplifier before the diode to boost signal. Further, Russell mentioned that some recorders don't like the AUX input connected during playback.

CONNECTICUT microCOMPUTER

150 POCONO ROAD - BROOKFIELD, CONNECTICUT 06804

(203) 775-9659





RS-232 PRINTER ADAPTER FOR THE COMMODORE PET

The CONNECTICUT microCOMPUTER ADApter model 1200 is the first in a line of peripheral adapters for the COMMODORE PET. The CmC ADA 1200 drives an RS-232 printer from the PET IEEE-488 bus. The CmC ADA 1200 allows the PET owner to obtain hard copy program listings, and to type letters, manuscripts, mailing labels, tables of data, pictures, invoices, graphs, checks, needlepoint patterns, etc., using a standard RS-232 printer.

The CmC ADA model 1200B comes assembled and tested, without power supplies, case, or RS-232 connector for $98.50. The CmC ADA 1200C comes complete for $169.00. Specify baud rate when ordering. (300 baud is supplied unless otherwise requested. Instructions for changing the baud rate are included.)

WORD PROCESSOR FOR THE COMMODORE PET

CONNECTICUT microCOMPUTER now has a word processor program for the COMMODORE PET. This program permits composing and printing letters, flyers, advertisements, manuscripts, articles, etc., using the COMMODORE PET and an RS-232 printer.

Script directives include line length, left margin, centering, and skip. Edit commands allow the user to insert lines, delete lines, move lines, change strings, save onto cassette, load from cassette, move up, move down, print and type.

The CmC Word Processor Program addresses an RS-232 printer through a CmC printer adapter.

The CmC Word Processor Program is available for $29.50.

RS-232 TO CURRENT LOOP/TTL ADAPTER

The CmC ADApter model 400 has two circuits. The first converts an RS-232 signal to a 20 ma current loop signal, and the second converts a 20 ma current loop signal to an RS-232 signal. With this device a computer's teletype port can be used to drive an RS-232 terminal, or vice versa, without modification of the port. The CmC ADA 400 can also be paralelled to drive a teletype or RS-232 printer while still using the computer's regular terminal. The CmC ADA 400 can easily be modified to become an RS-232 to TTL and TTL to RS-232 ADApter. The CmC ADA 400 does not alter the baud rate and uses standard power supplies. The current loop is isolated from the RS-232 signal by optoisolators.

The CmC ADA 400 is the perfect partner for KIM if you want to use an RS-232 terminal instead of a current loop teletype.

The CmC ADA 400S comes with drilled, plated through solder pads and sells for $24.50. The CmC ADA 400B comes with barrier strips and screw terminals and sells for $29.50.

This announcement was composed on a COMMODORE PET and printed on a GE TermiNet using a CmC ADA 1200C printer adapter and the CmC Word Processor Program.

Qnty	Description	baud rate	price	total
	CmC ADA 1200B (basic)		$98.50	
	CmC ADA 1200C (complete)		$169.00	
	CmC Word Processor Program (cassette)		$29.50	
	CmC ADA 400S (solder pads)		$24.50	
	CmC ADA 400B (barrier strips)		$29.50	
		Subtotal		
		Connecticut residents add 7% sales tax		
		Handling and shipping - add per order	$3.00	
		Foreign air mail - add $5.00 per order		
		Total included with order		

Mail with remittance or charge information to:

CONNECTICUT microCOMPUTER
150 Pocono Road, Room 6
Brookfield, Conn. 06804

NAME

COMPANY

ADDRESS

CITY

STATE ZIP

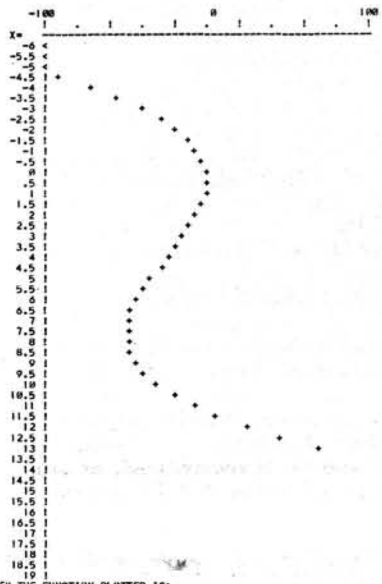
CHARGE TO: VISA MASTER CHARGE M/C INTERBANK NUMBER Expiration date

Credit card number

SIGNATURE

PET prints

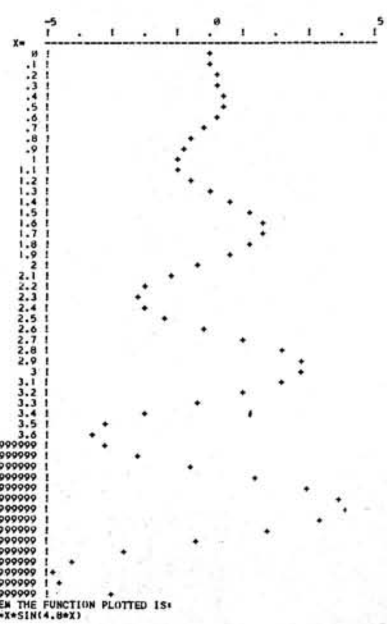
COMMODORE PET HARD COPY OUTPUT USING PET ADA 1200




```
1000 REM THE FUNCTION PLOTTED IS:
1010 Y=X*X*X/4-3*X*X+2*X-1
READY.
```

X	Y
0	0
.5	.33773159
1	-.996164609
1.5	1.1905018
2	-.348653564
2.5	-1.34143229
3	2.89697333
3.5	-3.10648462
4	1.37325972
4.5	1.71562719
5	-4.52789179

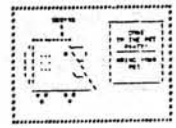
```
1000 REM THE FUNCTION PLOTTED IS:
1010 Y=X*SIN(4.8*X)
READY.
```

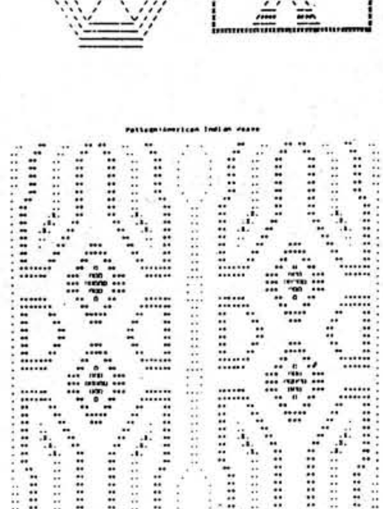


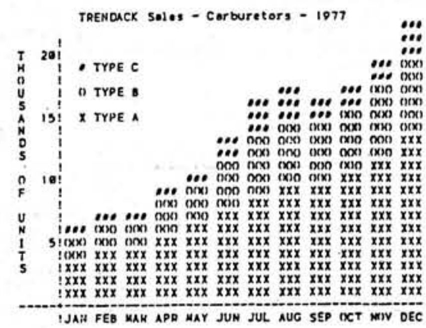

```
000 REM THE FUNCTION PLOTTED IS:
010 Y=X*SIN(4.8*X)
ADY.
```

```
1 OPEN 6,8:CMD 6:LIST
10 REM ARCSIN AND ARCCOS FUNCTIONS FOR THE COMMODORE PET
20 REM WRITTEN BY RICHARD ROSNER
30 REM              CONNECTICUT MICROCOMPUTER
40 REM              150 POCONO ROAD
50 REM              BROOKFIELD, CT 06804
60 REM
70 REM "Listed on a GE TermiNet 300
80 REM "using a CmC ADA 1200.
90 REM
100 REM OPEN OUTPUT FILE ON DEVICE #8.
110 OPEN 5,8
120 REM
500 REM GET A SINE VALUE
510 INPUT S
520 C=S
530 REM
1000 REM THE SINE OF THE ANGLE IS S
1010 REM IF THE SIN IS IN THE RANGE OF -1 TO 1, THEN COMPUTE.
1020 IF S<1 AND S>-1 GOTO 1050
1030 AS=90:GOTO2000
1040 REM THE ARCSINE IS AS
1050 AS=ATN(S/((1-S*S)^.5))
1060 REM THE RESULT IS IN RADIANS. CONVERT TO DEGREES.
1070 AS=AS*180/
2000 REM THE COSINE OF THE ANGLE IS C
2010 REM IF THE COSINE IS IN THE RANGE OF -1 TO 1,
2011 REM       AND NOT = TO 0, THEN COMPUTE
2020 IF C<>0 THEN 2040
2030 AC=90:GOTO 3000
2040 IF C<1 AND C>-1 THEN 2070
2050 AC=0:GOTO 3000
2060 REM THE ARCCOS IS AC
2070 AC=ATN((1-C*C)^.5/C)
2080 REM THE RESULT IS IN RADIANS. CONVERT TO DEGREES.
2090 AC=AC*180/
3000 PRINT#5," SIN,COS      ARCSIN       ARCCOS"
3010 PRINT#5,S;
3020 FOR N=1 TO 13-LEN(STR$(S)):PRINT#5," ";:NEXT
3030 PRINT#5,AS;
3040 FOR N=1 TO 13-LEN(STR$(AS)):PRINT#5," ";:NEXT
3050 PRINT#5,AC
3100 GOTO 500
READY.
```

SIN,COS	ARCSIN	ARCCOS
.1	5.73917048	84.2608296
SIN,COS	ARCSIN	ARCCOS
.2	11.536959	78.463041
SIN,COS	ARCSIN	ARCCOS
.3	17.4576031	72.5423969
SIN,COS	ARCSIN	ARCCOS
.33333333	19.4712204	70.5287796
SIN,COS	ARCSIN	ARCCOS
.4	23.5781785	66.4218216
SIN,COS	ARCSIN	ARCCOS
.5	30	60
SIN,COS	ARCSIN	ARCCOS
.6	36.8698977	53.1301024
SIN,COS	ARCSIN	ARCCOS
.7	44.427004	45.572996
SIN,COS	ARCSIN	ARCCOS
.72591002	46.5446035	43.4553965
SIN,COS	ARCSIN	ARCCOS
.8	53.1301024	36.8698977
SIN,COS	ARCSIN	ARCCOS
.9	64.1580673	25.8419328
SIN,COS	ARCSIN	ARCCOS
.999999	89.9189618	.0810382363





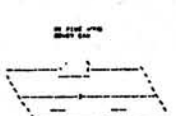


Powers of X

X	X^2	X^3	X^PI	X^5.25	X^.5
0	0	0	0	0	0
50	2500	125000	217506.221	830983735	7.07106781
100	10000	1000000	1919487.58	3.16227767E+10	10
150	22500	3375000	6861077.28	2.65753572E+11	12.2474487
200	40000	8000000	16939435.3	1.20339299E+12	14.1421356
250	62500	15625000	34146855.2	3.88315787E+12	15.8113883
300	90000	27000000	60548854.8	1.01131532E+13	17.3205081
350	122500	42875000	98271022.8	2.27173273E+13	18.7082869
400	160000	64000000	149490141	4.57946723E+13	20
450	202500	91125000	216427747	8.49895829E+13	21.2132034
500	250000	125000000	301345240	1.47772127E+14	22.3606798
550	302500	166375000	406540008	2.43727268E+14	23.4520788
600	360000	216000000	534342301	3.84852281E+14	24.4948974
650	422500	274625000	687112634	5.85861722E+14	25.4950976
700	490000	343000000	867239597	8.6449943E+14	26.4575131
750	562500	421875000	1.07713798E+09	1.24185712E+15	27.3861279
800	640000	512000000	1.31924718E+09	1.7426992E+15	28.2842713
850	722500	614125000	1.59602972E+09	2.39579338E+15	29.1547595
900	810000	729000000	1.90997007E+09	3.23424693E+15	30
950	902500	857375000	2.26357349E+09	4.29584822E+15	30.82207
1000	1000000	1E+09	2.6593656E+09	5.62341327E+15	31.6227766

Powers of X

X	X^2	X^3	X^PI	X^5.25	X^.5
0	0	0	0	0	0
50	2500	125000	217506.221	830983735	7.07106781
100	10000	1000000	1919487.58	3.16227767E+10	10
150	22500	3375000	6861077.28	2.65753572E+11	12.2474487
200	40000	8000000	16939435.3	1.20339299E+12	14.1421356
250	62500	15625000	34146855.2	3.88315787E+12	15.8113883
300	90000	27000000	60548854.8	1.01131532E+13	17.3205081
350	122500	42875000	98271022.8	2.27173273E+13	18.7082869
400	160000	64000000	149490141	4.57946723E+13	20
450	202500	91125000	216427747	8.49895829E+13	21.2132034
500	250000	125000000	301345240	1.47772127E+14	22.3606798
550	302500	166375000	406540008	2.43727268E+14	23.4520788
600	360000	216000000	534342301	3.84852281E+14	24.4948974
650	422500	274625000	687112634	5.85861722E+14	25.4950976
700	490000	343000000	867239597	8.6449943E+14	26.4575131
750	562500	421875000	1.07713798E+09	1.24185712E+15	27.3861279
800	640000	512000000	1.31924718E+09	1.7426992E+15	28.2842713
850	722500	614125000	1.59602972E+09	2.39579338E+15	29.1547595
900	810000	729000000	1.90997007E+09	3.23424693E+15	30
950	902500	857375000	2.26357349E+09	4.29584822E+15	30.82207
1000	1000000	1E+09	2.6593656E+09	5.62341327E+15	31.6227766



General Notes for Converting S-100 8K Memory Boards for PET
Prentice Orswell Boulder, CO

1. Logic for wait states and protect are not used.
2. Page select switches are not necessary.
3. Change MWRT driver on memory board so there are an even number of inversions or no inversions on the PET R/$\overline{W}$ line.
4. Use A0→A11 and:
   $\overline{SEL1}$ and $\overline{SEL2}$ for expansion of 4K PET.
   $\overline{SEL2}$ and $\overline{SEL3}$ for expansion of 8K PET.
   SEL(n) and $\overline{SEL}$ (n+1) for further expansion.
5. Parallel the memory board data input and output bus and connect to the PET BD0→BD7.
6. Bring R/$\overline{W}$ and the two PET select lines into the tri-state data bus driver logic so that the driver is only on when memory is selected and R/$\overline{W}$ is high.
7. Bring A10, A11 and the two select lines into the page select logic.
8. PET A0→A9 connect to the memory board A0→A9.
9. Pet memory expansion connector
   A27 RES
   A28 IRQ
   A29 B02 are not used for memory expansion
10. Get power from external power source and don't forget the ground connection to the PET (upper memory expansion connector pins).
11. Try pull-ups (560Ω) on the PET address, select, and R/$\overline{W}$ lines if you have problems.

Defeating "REDO FROM START"
Chuck Stuart Dallas, TX

The automatic printing done by PET BASIC when the operator goofs up the input such as "REDO FROM START", "EXTRA IGNORED" and "??" can really screw up a formatted screen layout.

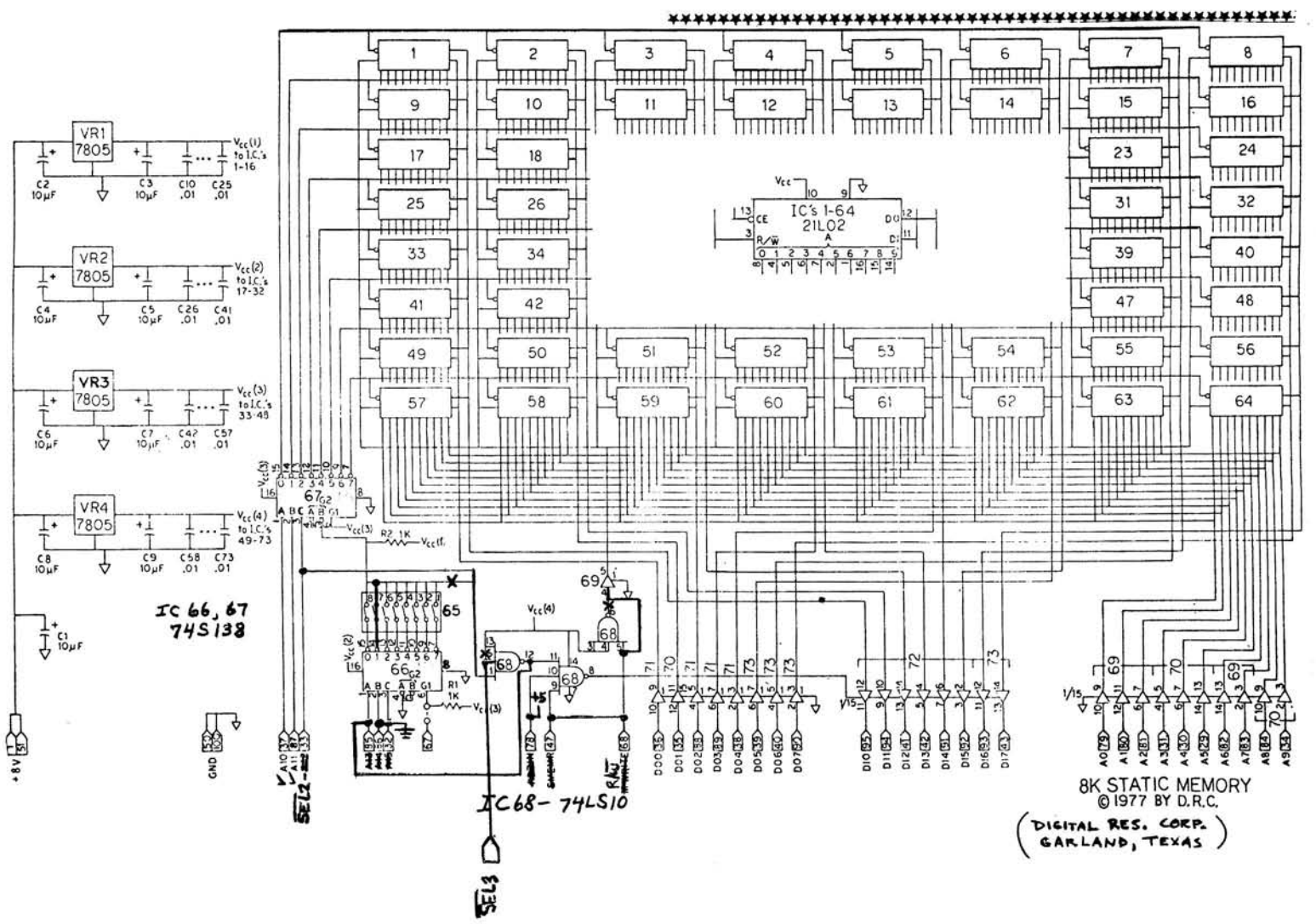
The most common error in inputting alpha data when numeric is expected. This will get you a 15 yard penalty and "REDO FROM START" every time.

One way to avoid this problem is to always input data from the keyboard in STR$ format, check it's validity in a subroutine, and then if everything is OK, convert the input to numeric with a VAL statement. The following subroutine can be used to perform this function.

```
10  INPUT A$ : GOSUB 100  IF F1=1 GOTO 10
20  ........
100 F1=0 : ND=0
110 FOR I=1 TO LEN(A$)
120 A1$=MID$(A$,I,1)
130 IF A1$<"0" OR A1$>"9" THEN GOTO 150
140 NEXT I : A=VAL(A$) : RETURN
150 IF A1$="-" AND I=1 GOTO 140
160 IF A1$="." AND ND=0 THEN ND=1: GOTO 140
170 F1=1: PRINT "cu";
180 PRINT A$; " IS NOT NUMERIC." : PRINT "cu";
190 RETURN
```

This subroutine checks each character in the string to see if it is a value between 0 and 9. If each individual character is a number then the string is numeric.

Line 150 allows a leading "-" to be accepted indicating a negative value. If all is OK, line 140 sets A to the numeric value of A$ and RETURNS. If a non-numeric character is encountered, an error message is printed and control is returned to the INPUT command.



PET Composite Video Interface
Cal E. Merritt Danville, Indiana

This circuit provides composite video output from the PET. I have used the output to drive two different monitors with good success. The circuit is very simple and can be put together with a wire wrap tool in a few minutes.

I used one of the existing PET 5 volt sources. The easiest way to steal the video and drives is to carefully scrape clean the fc

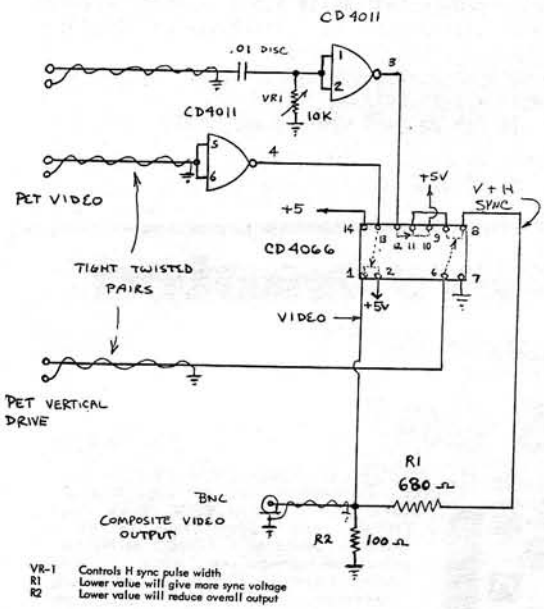
PET Composite Video Interface
Cal Merritt Danville, Indiana

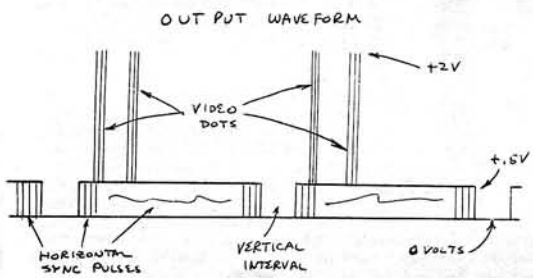
This circuit provides composite video output from the PET. I have used the output to drive two different video monitors with good success. The circuit is very simple and can be put together with a wire wrap tool in a few minutes.

I used one of the existing PET 5 volt sources. The easiest way to steal the video and drives is to carefully scrape clean the foils next to where the monitor plug is and tack solder a twisted pair to each signal and the closest ground buss. Other variations would work equally well.

To avoid metal shavings and such falling on the main board, I removed the back cover from the monitor (Power OFF) and mounted a BNC jack two inches to the right of the brightness control and fed it with a twisted pair. I mounted the board under one of the bolts that hold the monitor to the main chassis and attached the drive twisted pairs to the existing ones for the monitor.

Video monitors seem very tolerant and the two units I have used work fine. The only problem encountered was in attempting to do all white screen or very dense graphics which caused sync tear in one of the monitors. Normal or dense listings worked well.





All three monitors I tried worked with this video output. The appearance of the video will be a function of the quality of the monitor. Some of the scrapped out commercial units available with the 10 MHz and more bandwidths look excellent with the PET video. I have had a number of people comment that my 12" commercial monitor looks better than the built in unit. The add-on does not alter the existing PET display in any way.

PET SOFTWARE

CASINO SERIES:

These programs were developed both as a tutorial for those planning to visit a Casino and wanting to learn to play the game correctly, and as a means for the serious gambler to develop and thoroughly test a gambling system under actual Casino conditions without having to prematurely risk his capital. All betting odds and options available at the Casino of the MGM Grand Hotel in Las Vegas have been incorporated into these programs. Full screen graphics have been used to show the cards being dealt, the Roulette wheel spinning, and the dice being thrown in order to increase realism and highten enjoyment.

1. **Casino Blackjack:**
   Play with 1, 2, or 4 decks. Change decks or reshuffle at anytime. Bankroll keeps track of winnings (or losses). Split pairs, Double Down, or place an Insurance Bet. Full Casino rules and many other features. $9.95

2. **Casino Roulette:**
   Bet on one number, two numbers, three in a row, odd, black, etc. Watch the numbers come up as the wheel spins. Many unique features. Twelve different ways to bet. Spin the wheel. Where she stops, nobody knows. $9.95

3. **Casino Baccarat: (Chemin de Fir)**
   James Bond's favorite game. You get two games in one. Casino style and Blackjack style. Includes special features for help in developing a winning 'system'. The largest bets are usually at this table $9.95

4. **Casino Craps:**
   Eighty percent of the turnover at a Casino is at the crap tables. Bet the 'Do Line', 'Don't Line', Field or any of 10 different bets including Double Odds. This program alone can pay for your PET. $9.95

STRATEGIC BOARD SERIES:

For the young and old alike. Each game is played on one or more boards which are shown on the screen using PET graphics. You play against the PET. PET plays a fast exciting game, never taking over ten seconds to move. These games are beatable but you really have to try. Hours of entertainment.

1. **Qubic 4:**
   This is three dimensional Tic-Tac-Toe played on four 4x4 boards. 76 ways to win (or lose). PET chooses between three strategies for each game. No two alike $9.95

2. **Go-Moku:**
   This ancient Chinese board game is played with 20 men on a 9x9 board. You must get 5 men in a row either across, down, or diagonally in 30 move before PET does $9.95

QUANTITY DISCOUNTS:

Any two of the above programs $15.95. Any four $27.95. All six $39.95.

OUR GUARANTEE TO YOU:

If you feel any program is not as advertised, we will give you a 100% refund on that program. Any tapes that will not load will be replaced at no charge and any bugs will be promptly corrected. These programs were professionally designed and written and are as user proof as possible. Any incorrect input will generate a self explanatory error message. You are never left wondering what you did wrong or what to do next. We welcome your comments or suggestions on any of our software.

All orders shipped postage paid by 1st Class Mail immediately from stock.

Order From:
CMS, 5115 Menefee Drive, Dallas, Texas 75227

```
 0 GOTO 60000: REM UNLIST (LARRY TESLER)
 60000 OPEN 1,1: REM ***FILE SPEC***
 60010 GET#1,C$:IFASC(C$)<>13GOTO60010
 60020 POKE610,1:Y$="3":PRINTY$:D=-6:E=252:S=6:C=13:GET#1,C$:I
FC$="R"GOTO60050
 60030 PRINTC$;:IFASC(C$)<>CTHENGET#1,C$:S=D*((EANDST)=0):IFSG
OTO60030
 60040 Z=2:Z$="GOTO60020":IFSGOTO60070
 60050 CLOSE1:PRINTMID$("ERRORCONT",S+1,6);:STOP:PRINTY$:Z=9:Y
=60000:T=10
 60060 FORI=0TO7:PRINTY+T*I:NEXT:PRINT0
 60070 L=525:FORI=1TOZ+1:POKEL+I,C:NEXT:POKEL,Z:PRINTZ$:PRINT"
";:END
 61000 REM TO CONVERT AN ASCII PROGRAM
 61010 REM (SUCH AS PRODUCED BY
 61020 REM OPEN 1,1,1: CMD1: LIST
 61030 REM FORI=1TO200:PRINT#1," ";:NEXT
 61040 REM PRINT#1,"": CLOSE1)
 61050 REM TO BASIC, LOAD THIS PROGRAM
 61060 REM FIRST, THEN RUN, IT WILL
 61070 REM READ THE FIRST FILE ON
 61080 REM TAPE 1 AND ASSIMILATE EACH
 61090 REM LINE INTO THE PROGRAM.
 61100 REM FOR OTHER INPUT SOURCES,
 61110 REM CHANGE LINE 60000.
 61120 REM TO ADD A SUBROUTINE TO
 61130 REM A PROGRAM, FIRST LOAD
 61140 REM THIS UNLIST PROGRAM,
 61150 REM THEN LIST-61090,
 61160 REM THEN LOAD THE PROGRAM,
 61170 REM THEN HOME AND HIT RETURN
 61180 REM 9 TIMES TO PUT UNLIST
 61190 REM INTO THAT PROGRAM.
 61200 REM NEXT, PUT IN THE TAPE
 61210 REM THAT HAS AN ASCII VERSION
 61220 REM OF THE SUBROUTINE (MADE
 61230 REM BY LIST), AND RUN.
 61240 REM IF IT STOPS BY "BREAK",
 61250 REM YOU CAN DELETE THE UNLIST
 61260 REM CODE BY SAYING "CONT"
READY.
```

PET SOFTWARE

AMORTIZATION -compute your finances.....$6
BIORHYTHM-plotted before your eyes.....$7
BLACKJACK-full graphics version.....$7
DEPTHCHARGE-seek out enemy subs.....$7
HANGMAN -spelling game for children.....$8
MULTIPLICATION-learn without tables.....$7
TIC-TAC-TOE-in three dimensions.....$6

All programs are on ready-to-load digital cassette for 8K PETS. They are documented and easily modified. 20% discount on order of 3 or more. VISA/BA or MASTERCHARGE cards are accepted(include card# and expir.). Postage paid in U.S.

ABACUS SOFTWARE
P.O. BOX 7211
GRAND RAPIDS, MI. 49510

TNW488/232 Serial Interface

TNW 488/103 LOW SPEED MODEM

USE YOUR PET AS A TERMINAL

Now you can interface RS-232-C and MIL-STD-188C serial peripherals such as printers, modems, CRT terminals, plotters, paper tape readers and punches - even other computers - to your PET. The TNW488/232 Serial Interface provides two independent BIDIRECTIONAL asynchronous serial data channels for your PET, plus additional input and output signal lines you can use to control peripheral devices. Baud rate (75 to 9600 bits per second), character length, and parity are strapped on the board independently for the two channels. TNW488/232 Serial Interface assembled and tested: $280. Single channel version of TNW488/232: $240.

AUTO ORIGINATE, ANSWER, DIAL

The TNW488/103 Low Speed Modem is a full-service modem that puts you and your PET "on line". A PET program tape that allows you to use your PET as a terminal on a timesharing system accompanies the unit. Because the TNW488/103 connects to the telephone system through a CBT or CBS type Data Access Arrangement (not included), you can have your PET pulse-dial your phone calls, or answer calls from other computers. Baud rate (75 to 600 bits per second), character length, and parity are software selectable. TNW488/103 Low Speed Modem, assembled and tested: $320. Interface to CBT: standard; interface to CBS: $10 additional. Cable to DAA: $15

PLAY COMPUTER-TO-COMPUTER GAMES

HAVE YOUR COMPUTER DIAL YOUR CALLS

Both the TNW488/103 and the TNW488/232 incorporate a REAL IEEE 488 interface: other devices can be used on the bus with them, and they can be used with IEEE-capable computers other than PET. Each unit is a single 9 by 13 inch printed circuit board, with power supplies built right on the board: you provide only 117 VAC. An attractive cabinet is available ($35), as is a cable to connect to the PET's IEEE port ($20). An extensive documentation booklet accompanies each unit, and a 90 day limited warranty applies. California residents add 6% sales tax to all orders. We accept VISA and Mastercharge. Allow one to three weeks for delivery.

THE NET WORKS • 5924 QUIET SLOPE DRIVE • SAN DIEGO, CA 92120
Represented by
ASTRONICS • 4805 MERCURY STREET • SAN DIEGO, CA 92111 (714) 278-5441

FIND for PET
Jim Butterfield Toronto

Need to search for an expression, a variable, or a keyword? Slip program FIND in behind your program (it's not very long) - then insert a line 1 to say what to search for .. and the job's done. Every line in memory which contains the same expression as line 1 will be reported. This includes line 1 itself, of course, and any lines in program FIND .. as well as the program you're searching. I'm spacing out the program listing for readability -- close in the spaces when you input to save space.

```
9000  A=1025 : X=PEEK(1029) : FOR J=1 TO  1E3 : FOR K=A+4
      TO A+83
9001  P=PEEK(K) : IF P=X THEN GOSUB 9005
9002  IF P<>0 THEN NEXT K
9003  A=256*PEEK(A+1)+PEEK(A) : IF A>0 THEN NEXT J
9004  STOP
9005  FOR L=1 TO 80 : Y=PEEK(1029+L) : IF Y=0 THEN
       ? 256*PEEK(A+3)+PEEK(A+2); : RETURN
9006  IF Y=PEEK(K+L) THEN NEXT L
9007  RETURN
```

Example: to find all FOR statements in a program; insert FIND (above) and then insert line 1:

```
1  FOR
```

Now invoke FIND with RUN 9000. The program will print 1 followed by any program lines containing FOR followed by 9000 9000 9005. (9000 prints twice because it contains two FOR's).

FOR is a keyword, and doesn't store as three separate characters, so you wouldn't find it if you searched for characters FO. This can be handy: if you were looking for variable F you wouldn't get all the FOR's printed.

Modifications: if you squeezed P=0 just ahead of RETURN on line 9005 (it's a tight squeeze) a line number would print only once even when it had multiple matches; you might or might not want this feature.

Coding note: This is a type of structured program. Note that no loop can repeat without a limit. The complete absence of GOTO's is also considered characteristic of structured coding. A missing neatness element is that of line indentation; PET Basic just doesn't allow it.

IMPORTANT: Don't forget to wipe out line 1 and program FIND when you're finished with them.

```
**********************************************
1 REM TO ROUND A NUMBER N TO D DECIMAL PLACES, USE THE
2 REM STATEMENT IN LINE 30 BELOW:
3 REM
10 INPUT "D";D                          From
20 INPUT "N";N                          Roy Busdiecker
30 N=INT(N*10^D+0.5)/10^D
40 PRINT N
50 GOTO 20
60 REM
70 REM IF THE NUMBER OF PLACES DOES NOT CHANGE, REPLACE
80 REM 10^D WITH ITS VALUE.
```

Description of USR Function
Raynor Taylor Charleston, South Carolina

Probably the least documented and least understood element of the PET's BASIC is the USR function.

The USR function is concerned with the direct access to the processor and memory. The only other functions so concerned are PEEK, POKE, and SYS, which read memory, write memory, and call subroutines, respectively. In essence, USR is a combination of these three functions -- in fact, there is nothing USR can do that cannot be simulated through the use of these three.

Syntactically, USR is a pre-defined function, like SQR (square root) and INT (integer value of a number). For this reason, USR has numeric value and is used in calculations, assignment and print statements, etc. The only difference between USR and the others is that the value USR(X) takes on is not necessarily a function of the argument X.

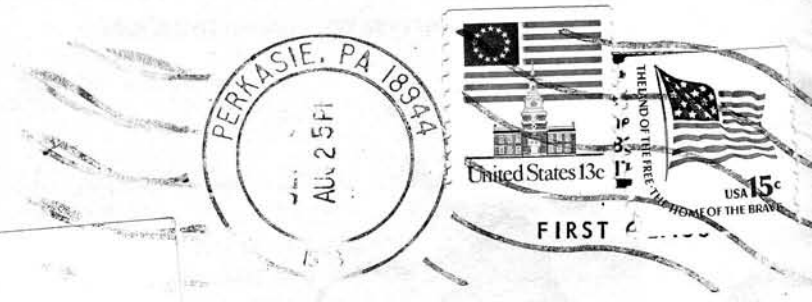
When USR is executed in a BASIC program, two things happen. First, the argument contained within the parenthesis is evaluated and stored in the PET's primary software-simulated accumulator $B1-$B5 under the standard five-byte binary representation. Second, the BASIC interpreter takes a jump to subroutine (JSR) to the location specified by the USR vector, memory locations $01 (low) and $02 (high) -- actually, an unconditional absolute jump instruction $4C is stored in $00, by the initialization routines upon reset, and the subroutine call is taken there.

After this is done, the machine language subroutine takes complete control of the PET. Although there are interrupts occurring all the time from various sources, the PET is at the disposal and mercy of the subroutine -- if something were to go wrong, the only way to bring the PET back to life would be to turn it off then back on, wiping out all programs in the process (the addition of the rese switch would be wise if you are planning on doing some machine language programming since turning the PET on and off is quite a strain on the CRT).

When the subroutine returns to BASIC ($CED2 for USR and $C6EE for SYS, incidentally) via the execution of a return from subroutine instruction (RTS) at the original stack level, interpretation of the BASIC program is continued as if nothing happened at all. The value that USR(X) assumes is the value of what is left in the five-byte primary accumulator -- if the subroutine changes nothing in this accumulator, then the value returned is what was put in the accumulator originally, namely the argument X.

```
**********************************************
1 REM TO LINE UP A COLUMN OF NUMBERS FOR OUTPUT, THE
2 REM STATEMENT IN LINE 140 BELOW MAY BE USED.
3 REM
110 K=20: REM -- MAX LENGTH             From
120 T=LOG(10)                           Roy Busdiecker
130 INPUT "N";N
140 PRINT SPC(K-INT(LOG(ABS(N))/T));N
150 GOTO 130
```

PET User Group
PO Box 371
Montgomeryville, PA 18936



James R. Schmidt
7331 W. Lyons St.
Morton Grove, IL 60053