$1

Washington Apple Pi

Volume 2 September 1980 Number 9

Highlights

FFT SUBROUTINE FOR THE 6502
BY Bruce F. Field

THE SHELL GAMES
BY Bruce Tognazzini
A REVIEW BY Tom Lucas

In This Issue

OFFICERS

President	-Bernard Urban	(301) 229-3458
Vice President	-Rich Wasserstrom	(703) 893-9147
Treasurer	-Bob Peck	(301) 468-2305
Secretary	-Dana Schwartz	(301) 725-6281
Members-at-Large	-Mark Crosby	(202) 488-1980
	-Sandy Greenfarb	(301) 674-5982
	-Hersch Pilloff	(301) 292-3100
Editor	-Bernard Urban	(above)
Associate Editor	-Rich Wasserstrom	(above)
	-Genevie Urban	(301) 229-3458
Librarian	-David Morganstein	(301) 972-4263

Washington Apple Pi
P. O. Box 34511
Washington, D.C. 20034
(301) 468-2305

Membership dues for Washington Apple Pi are $12.00 per calendar year. If you are interested in joining our club, call our number and leave your name and address. An application form will be mailed to you. Or if you prefer, write us at the above PO Box.

EVENT QUEUE

Washington Apple Pi meets on the 4th Saturday of each month at 9:30 AM at George Washington University School of Engineering, 23rd and H Streets, NW. Call the Club telephone during the week of the meeting for exact location and agenda.

NOVAPPLE meets on the 2nd Wednesday at 7:30 PM at Computers Plus in Franconia, and on the 4th Thursday at 7:30 PM at Computerland of Tysons Corner.

SIG-NEWS

The special interest group on games, SIGAMES, will hold its meeting at a location to be announced and immediately following the Washington Apple Pi monthly meeting. Sheldon Kramer will give a talk this month on shape tables.

DOS 3.3 BUGS

By Val Golding, CALL A.P.P.L.E.

We have heard from a couple of reliable sources that DOS 3.3 has been withdrawn to repair a couple of bugs, one of which could be major.

DOS 3.3 will run only on a 48K machine. There may be a problem in writing to random access files. Under certain conditions, it is said, DOS will blow the diskette. The current language resident in the Language Card will be wiped out on a reboot. The latter problem can be patched by writing directly to disk with an edit utility and changing byte CC of track 0 sector 9 to a 10. Thanks to Dave Holle of "Harvest" (Northwest Suburban Apple Group, Chicago) for this one.

We have no word on when it will be re-released. Apple, Inc. plans to make the correction available to early purchasers through its service centers.

EDITORIAL

We're back. Summer has come and gone too soon - oh, too soon... Thanks Rich, Bob, Dave and the rest for an excellent job. Sorry for the problems of changes in meeting places, poor or no air conditioning, etc. Hope this is all past us now....

Many changes are in the wind, some good and some not so good. First the bad. We will have to raise the dues. How much, I don't know yet. The good - strong talk of some sort of merger with maybe three other user groups. This could lead to more frequent and better organized sessions, more and better documented software. We are getting closer to offering a variety of courses. Our SIGs are taking hold. And our newsletter should improve with more articles and columns, fewer typos and, hopefully, checked-out software listings. Please note the changes in this issue. Thanks to Mark Crosby we will be able to use computer techniques to produce our headings, table of contents, etc. (See his article elsewhere in this issue). While I'm at it, anyone who has ideas on how the newsletter can be improved should see me at the meeting, call or drop a line, please. We should be coming out soon with the Best of Washington Apple Pi for the benefit of our newcomers who don't have all the back issues.

Say, folks, you have no excuse now. You may have missed the Apple Faire at N.Y.U. last August, but you can make amends by attending the Mid-Atlantic Computer Show. We'll be there! Come see us at Booth 452 and help "person" the table. (If you can spare the time and are willing to commit a time slot, please call me prior to September 18.) We will have the APPLE, some handouts, the cassettes of Woz and Jobs, and the latest issue of the Apple Orchard. We'll take new memberships and orders for disks and the Orchard.

MINUTES

EXECUTIVE BOARD MEETING

The Executive Board meeting of August 13, 1980, was held at Westat, Inc. in Rockville, and was called to order at 7:30 PM by the President.

Washington Apple Pi will be represented at the New York Computer Fair on August 16 by the President. Descriptive material promoting the club was shown in draft form. (In addition to the fair, this material will also be used in our own new member recruitment.) It was moved, seconded and passed that the Club partially reimburse the President for his trip to New York with $60.

It was moved, seconded and passed that the Club hire an instructor to teach a Beginning Basic course, charge members a nominal fee, and have the Club make up the difference, if any. The Membership

contd.

Chairman was instructed to design a Membership Card for the purpose of identifying members at meeting sales and discounts at participating local stores. The Treasurer announced he would no longer be able to coordinate group sales for the Club, and the Board will be actively seeking a replacement.

The Board discussed at great length the question of forming "Chapter" groups which would use the club services and pay dues to our parent organization, but operate independently. This would require amendment of the Club's Constitution and By-Laws. Negotiations will continue this month with officers of NOVAPPLE to determine the proper reimbursement for the Washington Apple Pi services which they currently receive at no charge. The Board is soliciting proposals from interested individuals on these and related matters.

The Board will recommend to the membership that dues be increased in 1981 to $18 per year to cover rapidly rising operational costs.

The meeting was adjourned at 9:45 PM.

GENERAL MONTHLY MEETING

The Washington Apple Pi meeting of August 23, 1980 was called to order at 9:35 AM by the Treasurer, with approximately 85 persons in attendance. It was announced that the Club would be sponsoring a class in Beginning Basic, and sufficient interest was shown by the members present.

John Moon unveiled his Apple Bulletin Board System (ABBS) and discussed its usage. The system will be available for WAP members exclusively. Volunteers were requested for the initial testing period.

Some of the membership expressed a desire for more information about Club operations. Motions were passed to republish the Constitution and By-Laws in the newsletter and to have a verbal Treasurer's report at each monthly meeting. Membership Chairman Tom Jones acknowledged that a proposal for the membership list to be distributed to Club members will be published in a forthcoming newsletter.

The meeting was then turned over to Sandy Greenfarb for a review of his recent experiences with the new DOS 3.3.

Dana J. Schwartz, Secretary

CLASSIFIEDS

For Sale: 48K APPLE II with AS Card; DOS 3.2 and DOS 3.3; DC Hayes Modem; valuable software (including APPLE WRITER); plus publications and manuals; 9" B & W TV if needed. Hardware value $2694.80, asking $2200, or make offer. Scooter Conrad, (301) 725-6251 after 4PM.

SCROLL DOWN

BY Dana J. Schwartz

The APPLE's Monitor makes it easy for the programmer to add text lines at the bottom of the screen and scroll up the rest of the screen. However, if you want to do the opposite, i.e. add a line at the top of the screen and scroll down the rest, you face a difficult problem.

Recently I have seen Assembly Language subroutines which will scroll downwards, but I have found it inconvenient to develop, and difficult to maintain, a Basic program which contains them (or any Machine Language, for that matter). The subroutine shown below (lines 1000-1040) performs the scroll down function, is written entirely in Integer Basic, and is only slightly slower that an Assembler version.

The detailed operation is left as an exercise for the interested student, but note that a 1024-byte buffer must be supplied (low and high address bytes in BL and BH, respectively). In line 10 this buffer is placed just below the program.

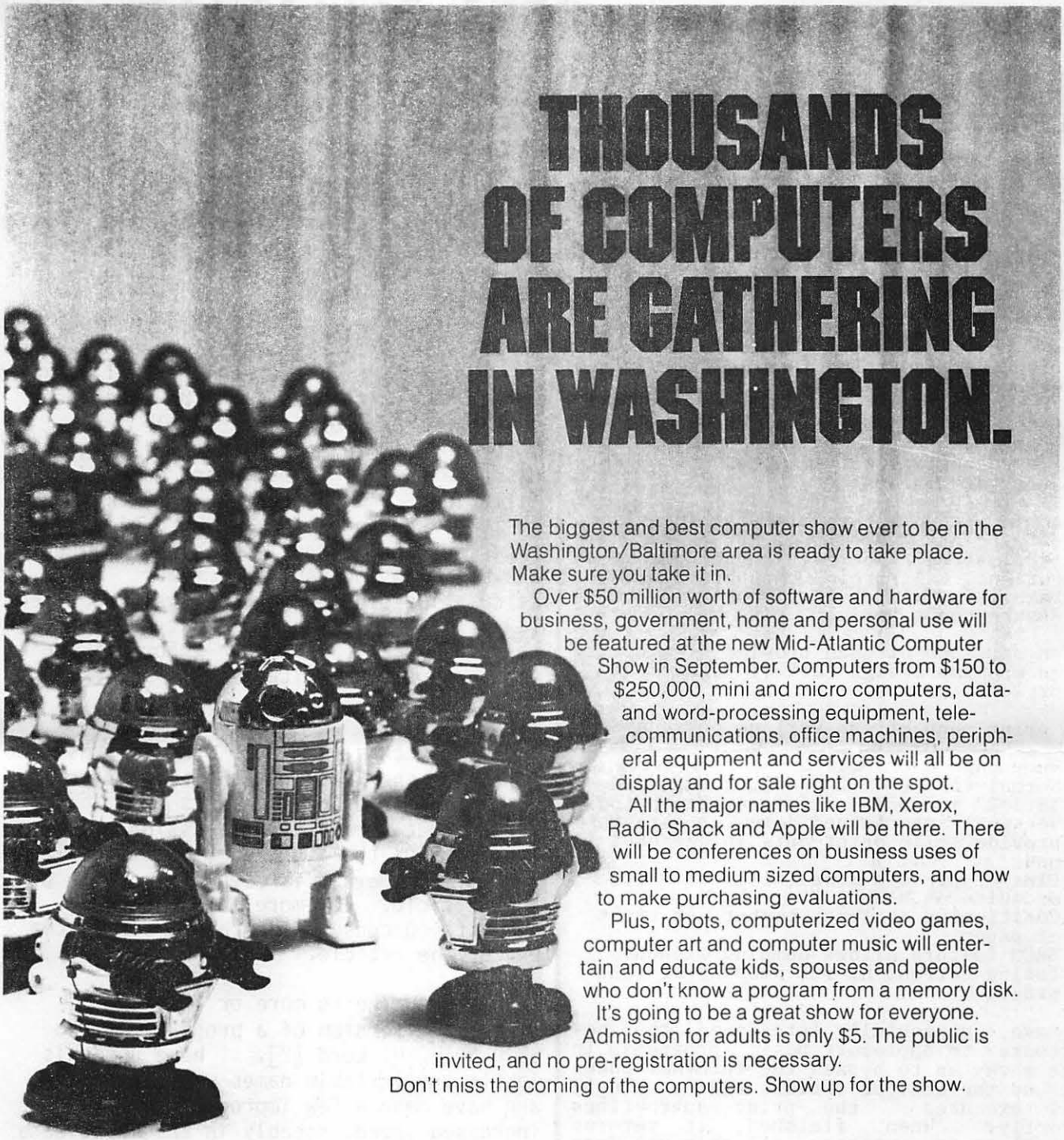
It should also be noted that some of the Peripheral Slot Scratchpad RAM locations (see pages 82-83 of the new Reference Manual) are altered by this subroutine.

```
]LIST
   10 BL=0:BH= PEEK (203)-4:I=0
   20 GOSUB 1000: REM SCROLL DOWN
   30 TAB 20-I MOD 20: PRINT "*";
      : TAB 20+I MOD 20: PRINT "*"
      : REM NEW TOP LINE
   40 I=I+1: IF PEEK (-16384)<128
       THEN 20: REM REPEAT UNTIL KEYIN

   50 POKE -16368,0: VTAB 24: END

  900 REM ****************************
  910 REM *                          *
  920 REM *       SCROLL DOWN        *
  930 REM *                          *
  940 REM *   BY DANA J. SCHWARTZ    *
  950 REM *                          *
  960 REM *         7/23/80          *
  970 REM *                          *
  980 REM ****************************
 1000 POKE 60,0: POKE 61,4: POKE
      62,207: POKE 63,7: POKE 66,
      BL: POKE 67,BH: CALL -468
 1010 POKE 60,BL: POKE 61,BH: POKE
      62,BL+119: POKE 63,BH+3: POKE
      66,128: POKE 67,4: CALL -468

 1020 POKE 60,BL+128: POKE 61,BH+
      3: POKE 62,BL+207: POKE 63,
      BH+3: POKE 66,40: POKE 67,4
      : CALL -468
 1030 VTAB 1: TAB 1: CALL -868
 1040 RETURN
```



THOUSANDS OF COMPUTERS ARE GATHERING IN WASHINGTON.

The biggest and best computer show ever to be in the Washington/Baltimore area is ready to take place. Make sure you take it in.

Over $50 million worth of software and hardware for business, government, home and personal use will be featured at the new Mid-Atlantic Computer Show in September. Computers from $150 to $250,000, mini and micro computers, data- and word-processing equipment, telecommunications, office machines, peripheral equipment and services will all be on display and for sale right on the spot.

All the major names like IBM, Xerox, Radio Shack and Apple will be there. There will be conferences on business uses of small to medium sized computers, and how to make purchasing evaluations.

Plus, robots, computerized video games, computer art and computer music will entertain and educate kids, spouses and people who don't know a program from a memory disk.

It's going to be a great show for everyone. Admission for adults is only $5. The public is invited, and no pre-registration is necessary.

Don't miss the coming of the computers. Show up for the show.

WASHINGTON, D.C.
D.C. ARMORY/STARPLEX, THURSDAY-SUNDAY, SEPTEMBER 18-21
11 A.M. TO 9 P.M. THURS.-SAT., 11 A.M. TO 5 P.M. SUN.

THE MID-ATLANTIC COMPUTER SHOW

Produced by National Computer Shows, 824 Boylston Street, Chestnut Hill, MA 02167, Telephone (617) 739-2000.

ENHANCED PAPER TIGER GRAPHICS SOFTWARE

A REVIEW BY Mark L. Crosby

If you own an IDS 440 "Paper Tiger" printer and you have a need to dump high-resolution pages 1 or 2 to paper, this is for you!

The original version of this newer, and much more versatile utility, was somewhat difficult to deal with. It requires a machine-language subroutine located at an inconvenient place (just below hi-res page 2). The Basic code required to activate it is also complex and difficult to interface with other Basic programs.

Computer Station of Illinois has really come through by announcing an excellent, and apparently bug-free, product. Here are the particulars:

Usable with all the following interface cards - APPLE Parallel, APPLE Centronics, APPLE Serial, APPLE Communications, California Computers Systems Parallel, California Computers Systems Asynchronous Serial

Two versions of code provide flexibility in program design (either located hi or lo in memory)

A print menu offers various options:

Dump page 1 or page 2
Normal (4 3/8" x 2 3/4") or expanded (6 1/2" x 5 3/8") print-out dimensions
Vertical interleaved dots (or not) to provide solid print-outs in expanded mode
Plot or picture mode (black on white or white on black)
Positioning at left, center, or right of paper
BRUN feature allows dumping without losing current hi-res display or your program

I have successfully interfaced the new software to Applesoft Basic programs in such a way as to bypass the internal pokes set up the desired options and a single call executes the print subroutines directly. When finished, it returns control to the program. All this was done without any modification to their software.

If you purchased the older version, you may now update it for only $10.00. It can be purchased for $44.95 at most local stores or by adding $2.00 shipping and handling and ordering from:

Enhanced Paper Tiger Graphics Software
$44.95
#7541 Applesoft
#7505 Pascal

COMPUTER STATION
12 CROSSROADS PLAZA
GRANITE CITY, IL 62040
(618) 452-1860

FFT SUBROUTINE FOR THE 6502

BY Bruce F. Field

The recent revolution in digital electronics has lead to a switch from analog circuitry to digital circuitry for many signal processing applications. Music and speech analysis are two areas that can now be reasonably pursued by the home experimenter. The basis for the analysis of sampled data is often the determination of the frequency response samples by use of the discrete Fourier transform (DFT). Performing calculations in the frequency domain often allows considerable insight into a problem, as well as permitting discrete-time convolution and hence digital filtering. The DFT became a practical tool in 1965 when Cooley and Tukey published a paper presenting an algorithm for computing the DFT with substantially fewer calculations than were previously required [1]. This algorithm, the fast Fourier transform (FFT), is now an important and popular tool in digital signal processing.

The 6502 machine language program presented here uses the FFT algorithm to compute the discrete Fourier coefficients of a 256 point input series. A discussion of the mechanics of the algorithm, or of Fourier series in general, is beyond the scope of this article. For more information I suggest you consult the references at the end of the article.

This subroutine is more or less a translated version of a program for the 6800 by R. H. Lord [2]. I have used his labels and variable names where possible and have made a few improvements for increased speed, notably in the input data set-up, the pre-transform bit swap, and the multiply routine. I urge you to get a copy of Lord's article for more detailed information.

The input data for the subroutine is a 256 byte block of 8 bit two's complement data which is stored at memory locations $4000 to $40FF ($ indicates a hexadecimal number). After transformation the cosine (or real) coefficients are stored in the table "REAL" from $4100 to $41FF and the sine (or imaginary) coefficients are stored in the table "IMAG" from $4200 to $42FF. The coefficients are stored in straight numerical order, symmetrically reflected

contd.

about the center of the table. The first entry in each array represents the dc component of the input. The second entry represents a sine wave with a period equal to the duration of the 256 input samples. Each remaining element represents a multiple of this frequency with the 129th element corresponding to the 128th harmonic. The remainder of the array is symmetric about the 129th element with the 130th element corresponding to the 127th harmonic, etc. The dc component would be the 257th element and therefore does not reappear and the sign of the imaginary data is reversed in the second half of the array.

All calculations are done in fixed point for greater speed and the data is scaled (divided by 2) as necessary to prevent overflow during the calculations. The number of scaling operations performed is stored in "SCLFCT" (memory location $B).

I have also included a short Applesoft program to demonstrate how the FFT subroutine works. The program generates an input waveform which is a combination of 3rd, 5th and 25th order harmonics, and plots the waveform. The FFT subroutine is called and then the magnitude of the transformed data is plotted in bar graph form. The leftmost bar is the dc component, followed by the fundamental, and harmonics up to the 128th. Everything is straightfoward except for line 340 which converts the value of 'A' to an 8 bit two's complement number, and lines 530 and 540 which convert 'A' and 'B' from two's complement to signed magnitude for plotting. After running the program you will notice some small values for other harmonics that should be equal to zero. Since the input components are all exact multiples of the sampling frequency (i.e. 256 samples equal an integral number of signal periods), the residuals are due to roundoff errors. If these errors are not tolerable the only solution is to re-write the subroutine with higher precision numbers (more than 8 bits) and settle for slower operation.

A word of warning, the transform results are correct only when the 256 samples span an exact multiple of the input frequency. If this is not possible, the more periods included in the 256 samples the greater the accuracy will be, but at the expense of restricting the number of harmonics that can be analyzed. The problems of leakage (sampling over a non-integral number of periods) and aliasing (not sampling at more than twice the highest input frequency) can create serious inaccuracies in the results. Several of the referenced texts deal with these problems rather thoroughly.

Execution time for the FFT subroutine is approximately 1.3 seconds which unfortunately is not fast enough for any real time data processing, but it is at least faster than the 2 or 3 minutes BASIC would take.

I have submitted this program to the club library so hopefully no one will have to type it in, but if you do, note that there is a table of cosine values at $4300 to $43FF that must be included. In order to increase the speed of the program these values are stored as signed magnitude binary with the eighth (sign) bit set for negative values.

REFERENCES

1. J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series", Math. Comput., vol. 19, pp.297-301, Apr. 1965.

2. R. H. Lord, "Fast Fourier for the 6800", Byte, vol. 4, pp. 108-118, Feb. 1979.

3. S. D. Stearns, Digital Signal Analysis, Hayden Book Co. Inc., Rochell Park NJ, 1975.

4. L. R. Rabiner and B. Gold, Theory and Applications of Digital Signal Processing, Prentice-Hall, Inc., Englewood NJ, 1975.

5. B. Gold and C. M. Rader, Digital Processing of Signals, McGraw-Hill, New York NY, 1969.

6. G. D. Bergland, "A Guided Tour of the Fast Fourier Transform", IEEE Spectrum, vol. 6, pp. 41-52, July 1969.

7. W. T. Cochran et al., "What is the fast Fourier transform?", IEEE Trans. Audio and Electroacoustics, vol. AU-15, pp. 45-55, June 1967.

8. R. Bracewell, The Fourier Transform and Its Applications, McGraw-Hill, New York NY, 1965.

contd.

```
]LOAD FFT TEST
]LIST

100  REM COMPUTES A COMBINATION
110  REM OF 3 SINUSOIDS, 3RD,
120  REM 5TH, AND 25TH HARMONICS
130  REM
140  REM THE PROGRAM THEN
150  REM CALCULATES THE FFT AND
160  REM PLOTS THE MAGNITUDE
170  REM OF THE COEFFICIENTS
175  REM B.F. FIELD MAY 80
180  REM
190  PRINT  CHR$ (4);"BLOAD FFT,A$4300"
200  HOME
210 W = 3.14159265 / 128
220  HGR : HCOLOR= 3
230  REM FRAME PLOTTING AREA
240  HPLOT 0,144 TO 255,144 TO 255,16 TO 0,16 TO 0,144
250  HPLOT 0,80 TO 255,80
255  REM WAVEFORM GENERATION
260 DC = 0:A0 = 30:A1 = 60:A2 = 15
270  VTAB 24: PRINT "INPUT WAVEFORM"
280  HPLOT 0,80 - DC / 2
290  REM COMPUTE SIGNAL AND PLOT
300  FOR I = 0 TO 255
310 WI = W * I
320 A = DC + A0 *  SIN (3 * WI) + A1 *  SIN (5 * WI) + A2 *  SIN
     (25 * WI)
330  HPLOT  TO I,80 - A / 2
340  IF A < 0 THEN A = 256 + A
345  REM STORE DATA IN INPUT ARRAY
350  POKE 16384 + I, INT (A + .5)
360  NEXT I
370  REM CALL FFT SUBROUTINE
380  PRINT : PRINT : PRINT "CALCULATING FFT"
390  CALL 17408
400  PRINT : PRINT : PRINT "   FREQUENCY SPECTRUM  -  MAGNITUDE"
410  REM PLOT MAGNITUDE OF
420  REM COEFFICIENTS
430  HGR
440  HPLOT 9,130 TO 267,130 TO 267,2 TO 9,2 TO 9,130
450  FOR I = 130 TO 10 STEP  - 10
460  HPLOT 3,I TO 7,I: NEXT
470  FOR I = 0 TO 128
480 M = I + 16896: REM  IMAG
490  REM GET REAL AND IMAG PART
500  REM CONVERT 2'S COMPLEMENT
510  REM TO SIGNED MAGNITUDE
520 A =  PEEK (M):B =  PEEK (M - 256)
530  IF B > 127 THEN B = B - 256
540  IF A > 127 THEN A = A - 256
550 A =  SQR (A * A + B * B)
560 K = I * 2 + 10
570  HPLOT K,130 TO K,130 - A
580  NEXT
590  END

]
```


contd.

```
LOAD FFT.SRC
:ASM
```

```
1000 *
1010 *       ----------------------------------
1020 *        FAST FOURIER TRANSFORM SUBROUTINE
1030 *                  FOR THE 6502
1040 *       ----------------------------------
1050 *
1060 *       ORIGINAL VERSION FOR THE 6800 BY
1070 *       R.H. LORD, "FAST FOURIER FOR THE 6800"
1080 *       BYTE, VOL. 4, PP. 108-118, FEB. 79.
1090 *
1100 *       LOOSELY TRANSLATED WITH SPEED
1110 *       IMPROVEMENTS FOR THE 6502 BY
1120 *       B.F. FIELD, APRIL 80.
1130 *
1140 *
1150 * THIS SUBROUTINE PERFORMS A 256 FFT
1160 * ON THE DATA IN THE INPUT DATA TABLE.
1170 * INPUT DATA IS ASSUMED TO BE TWO'S COMPLEMENT.
1180 * THE SUBROUTINE GENERATES A COSINE (REAL) AND SIN
1190 * (IMAGINARY) DATA TABLE AT "REAL" AND "IMAG"
1200 * THE RESULTANT TRANSFORM DATA IS 128 POINTS
1210 * SYMMETRICALLY REFLECTED ABOUT THE CENTER OF
1220 * THE 256 POINT TABLE.
1230 *
1240 * THE SUBROUTINE ASSUMES THAT THE INPUT DATA
1250 * IS ALL REAL AND THEREFORE DOES NOT MANIPULATE
1260 * THE IMAGINARY PORTION UNTIL AFTER THE FIRST
1270 * PASS.
1280 *
1290 * THE SUBROUTINE SCALES THE DATA WHENEVER
1300 * IT ANTICIPATES OVERFLOW.  THE SCALE FACTOR
1310 * COUNT IS AVAILABLE IN "SCLFCT".
1320 *
1330 *       THE SINE TABLE IS LOADED WITH THE PROGRAM.
1340 *       THE TABLE IS ACTUALLY A COSINE WAVE
1350 *       STORED AS 8 BIT SIGNED MAGNITUDE
1360 *       BINARY.
1370 *
1380 *
1390 *       ------------
1400 *        DATA AREAS
1410 *       ------------
1420 *
1430 INPUT  .EQ $4000      INPUT DATA
1440 REAL   .EQ $4100        REAL OUTPUT
1450 IMAG   .EQ $4200        IMAG OUTPUT
1460 SINE   .EQ $4300        SINE TABLE
1470 *
1480 *       ---------------------
1490 *        BASE PAGE VARIABLES
1500 *       ---------------------
1510 *
1520 MCD    .EQ 6             MULTIPLICAND
1530 MPR    .EQ 8             2'S COMP MULTIPLIER
1540 RES    .EQ 9             TEMPORARY RESULT
1550 SCLFCT .EQ $B            SCALE FACTOR CTR.
1560 CELNUM .EQ $C            CELLS FOR THIS PASS
```


contd.

```
                    1570 DELTA  .EQ $D          ANGLE INCREMENT
                    1580 PAIRNM .EQ $E          PAIRS/CELL
                    1590 PAIRCT .EQ $F          PAIR COUNTER
                    1600 CELDIS .EQ $10         CELL OFFSET
                    1610 CELCT  .EQ $11         CELL COUNTER FOR PASS
                    1620 PTRM   .EQ $12         "M" POINTER
                    1630 PTRN   .EQ $13         "N" POINTER
                    1640 SINPT  .EQ $14         SINE TABLE POINTER
                    1650 SINA   .EQ $15         TEMPORARY SINE
                    1660 COSA   .EQ $16         TEMPORARY COSINE
                    1670 TREAL  .EQ $17         TEMP. REAL DATA
                    1680 TIMAG  .EQ $18         TEMP. IMAG DATA
                    1690 TM     .EQ $19         WORKING STORAGE
                    1700 *
                    1710        .OR $4400
                    1720 *
                    1730 *      ------------------------------
                    1740 *       INPUT DATA SETUP AND BIT SWAP
                    1750 *      ------------------------------
                    1760 *
                    1770 *      TAKES REAL DATA FROM INPUT ARRAY
                    1780 *      PERFORMS PRE-TRANSFORM BIT SWAP
                    1790 *      AND STORES IT IN REAL ARRAY
                    1800 *      ALSO ZEROES IMAG ARRAY
                    1810 *
4400- A2 00         1820        LDX #0          INIT LOOP COUNTER
4402- 86 0B         1830        STX SCLFCT      ZERO SCALING FACTOR
4404- A0 08         1840 BITREV LDY #8          INIT BIT REV COUNTER
4406- 86 19         1850        STX TM          TEMP FOR X
4408- 66 19         1860 BRV1   ROR TM          ROTATE INTO CARRY
440A- 2A            1870        ROL             AND INTO ACCUM
440B- 88            1880        DEY             DECREM COUNTER
440C- D0 FA         1890        BNE BRV1        SEE IF DONE
440E- 86 19         1900        STX TM          RESTORE PTR FOR CMP
4410- C5 19         1910        CMP TM          IF ORIG>BIT REV DON'T
4412- 90 15         1920        BCC BRV2        DO SWAP
4414- A8            1930        TAY             BIT REV TO Y
4415- BD 00 40      1940        LDA INPUT,X     GET VALUE
4418- 99 00 41      1950        STA REAL,Y      INTO ARRAY
441B- B9 00 40      1960        LDA INPUT,Y
441E- 9D 00 41      1970        STA REAL,X
4421- A9 00         1980        LDA #0
4423- 99 00 42      1990        STA IMAG,Y      ZERO IMAG ARRAY
4426- 9D 00 42      2000        STA IMAG,X
4429- E8            2010 BRV2   INX             INCREM LOOP COUNTER
442A- D0 D8         2020        BNE BITREV
                    2030 *
                    2040 *      ---------------
                    2050 *       FFT FIRST PASS
                    2060 *      ---------------
                    2070 *
                    2080 *      SINCE IN PASS 1 ALL ANGLES ARE
                    2090 *      MULTIPLES OF 180 DEG. THERE ARE NO
                    2100 *      PRODUCT TERMS AND NO IMAG TERMS
                    2110 *      FAST VERSION OF PASS 1
                    2120 *
442C- 20 15 45      2130        JSR SCALE       SCALE IF ANY
                    2140 *                      OVER-RANGE DATA
442F- A2 00         2150        LDX #0          LOOP COUNTER
4431- 38            2160 PASS1  SEC             SET CARRY FOR SUB
```


contd.

```
4432- BD 00 41 2170          LDA REAL,X      GET RM
4435- FD 01 41 2180          SBC REAL+1,X    RM-RN
4438- 85 19    2190          STA TM          SAVE TEMP
443A- 18       2200          CLC             CLEAR CARRY FOR ADD
443B- BD 01 41 2210          LDA REAL+1,X    GET RN
443E- 7D 00 41 2220          ADC REAL,X      RM+RN
4441- 9D 00 41 2230          STA REAL,X      RM'=RM+RN
4444- A5 19    2240          LDA TM          GET RM-RN
4446- 9D 01 41 2250          STA REAL+1,X    RN'=RM-RN
4449- E8       2260          INX             INCREM LOOP COUNTER
444A- E8       2270          INX             TWICE
444B- D0 E4    2280          BNE PASS1       SEE IF DONE
               2290 *
               2300 *        --------------------
               2310 *         COMPUTATION OF FFT
               2320 *          PASSES 2 THRU N
               2330 *        --------------------
               2340 *
444D- A9 40    2350          LDA #64         SET UP PARAMETERS
444F- 85 0C    2360          STA CELNUM      FOR CELL COUNT
4451- 85 0D    2370          STA DELTA       ANGLE INCREMENT
4453- A9 02    2380          LDA #2
4455- 85 0E    2390          STA PAIRNM      PAIRS/CELL
4457- 85 10    2400          STA CELDIS      DISTANCE BETWEEN PAIRS
               2410 *
4459- 20 15 45 2420 NPASS    JSR SCALE       CHECK FOR OVERFLOW
445C- A5 0C    2430          LDA CELNUM      GET NUMBER OF CELLS
445E- 85 11    2440          STA CELCT       PUT IN COUNTER
4460- A2 00    2450          LDX #0
4462- 86 12    2460          STX PTRM        INIT DATA POINTERS
4464- 86 13    2470          STX PTRN
4466- A2 00    2480 NCELL    LDX #0
4468- 86 14    2490          STX SINPT       SET SINE POINTER
446A- A5 0E    2500          LDA PAIRNM      SET PAIR COUNTER
446C- 85 0F    2510          STA PAIRCT
446E- 18       2520 NC1      CLC
446F- A5 12    2530          LDA PTRM
4471- 65 10    2540          ADC CELDIS      COMPUTE PTRN
4473- 85 13    2550          STA PTRN
4475- A6 14    2560          LDX SINPT       GET SINE POINTER
4477- BD 00 43 2570          LDA SINE,X      GET COSINE
447A- 85 16    2580          STA COSA        SAVE IT
447C- BD 40 43 2590          LDA SINE+64,X   GET SINE
447F- 85 15    2600          STA SINA        SAVE IT
               2610 *
4481- A6 13    2620          LDX PTRN
4483- 85 06    2630          STA MCD         SAVE SIN FOR MPY
4485- BD 00 41 2640          LDA REAL,X      GET RN
4488- 20 57 45 2650          JSR MPY         RN*SIN(A)
448B- 85 18    2660          STA TIMAG       SAVE IT
448D- A5 16    2670          LDA COSA        GET COS
448F- 85 06    2680          STA MCD         SAVE FOR MPY
4491- BD 00 41 2690          LDA REAL,X      GET RN
4494- 20 57 45 2700          JSR MPY         RN*COS(A)
4497- 85 17    2710          STA TREAL
4499- A5 15    2720          LDA SINA        GET SIN
449B- 85 06    2730          STA MCD         SAVE IT
449D- BD 00 42 2740          LDA IMAG,X      GET IN
44A0- 20 57 45 2750          JSR MPY         IN*SIN(A)
44A3- 18       2760          CLC                                  contd.
```

```
44A4- 65 17     2770          ADC TREAL       TR=RN*COS+IN*SIN
44A6- 85 17     2780          STA TREAL
44A8- A5 16     2790          LDA COSA        GET COS
44AA- 85 06     2800          STA MCD         SAVE IT
44AC- BD 00 42  2810          LDA IMAG,X      IN
44AF- 20 57 45  2820          JSR MPY         IN*COS(A)
44B2- 38        2830          SEC
44B3- E5 18     2840          SBC TIMAG       TI=IN*COS-RN*SIN
44B5- 85 18     2850          STA TIMAG
                2860 *
44B7- 18        2870          CLC
44B8- A6 12     2880          LDX PTRM        GET M POINTER
44BA- BD 00 41  2890          LDA REAL,X      RM
44BD- 85 19     2900          STA TM          SAVE TEMP
44BF- 65 17     2910          ADC TREAL
44C1- 9D 00 41  2920          STA REAL,X      RM'=RM+TR
44C4- 38        2930          SEC
44C5- A5 19     2940          LDA TM
44C7- E5 17     2950          SBC TREAL
44C9- A6 13     2960          LDX PTRN
44CB- 9D 00 41  2970          STA REAL,X      RN'=RM-TR
44CE- 18        2980          CLC
44CF- A6 12     2990          LDX PTRM        M POINTER
44D1- BD 00 42  3000          LDA IMAG,X      IM
44D4- 85 19     3010          STA TM          SAVE TEMP
44D6- 65 18     3020          ADC TIMAG
44D8- 9D 00 42  3030          STA IMAG,X      IM'=IM+TI
44DB- 38        3040          SEC
44DC- A5 19     3050          LDA TM          IM
44DE- E5 18     3060          SBC TIMAG
44E0- A6 13     3070          LDX PTRN
44E2- 9D 00 42  3080          STA IMAG,X      IN'=IM-TI
                3090 *
                3100 *        INCREMENT POINTERS FOR NEXT PAIR
                3110 *
44E5- 18        3120          CLC
44E6- A5 14     3130          LDA SINPT       SINE POINTER
44E8- 65 0D     3140          ADC DELTA       ADD ANGLE INCREM
44EA- 85 14     3150          STA SINPT
44EC- E6 12     3160          INC PTRM        INCREM M POINTER
44EE- E6 13     3170          INC PTRN        INCREM N POINTER
44F0- C6 0F     3180          DEC PAIRCT      DEC NUMBER OF PAIRS
44F2- F0 03     3190          BEQ NXTCEL      GO TO NEXT CELL
44F4- 4C 6E 44  3200          JMP NC1         CONTINUE
                3210 *
                3220 *        DO NEXT CELL
                3230 *
44F7- 18        3240 NXTCEL   CLC
44F8- A5 12     3250          LDA PTRM        TO M POINTER
44FA- 65 10     3260          ADC CELDIS      ADD CELDIS
44FC- 85 12     3270          STA PTRM
44FE- C6 11     3280          DEC CELCT       DECREM CELL COUNTER
4500- F0 03     3290          BEQ NP1         THIS PASS DONE IF 0
4502- 4C 66 44  3300          JMP NCELL       CONTINUE
                3310 *
                3320 *        CHANGE PARAMETERS FOR NEW PASS
                3330 *
4505- 46 0C     3340 NP1      LSR CELNUM      HALF AS MANY CELLS
4507- F0 09     3350          BEQ DONE        NO MORE CELLS?
4509- 06 0E     3360          ASL PAIRNM      TWICE AS MANY PAIRS
```

contd.

```
450B- 06 10     3370          ASL CELDIS       TWICE AS FAR APART
450D- 46 0D     3380          LSR DELTA        HALF THE ANGLE
450F- 4C 59 44  3390          JMP NPASS        DO NEXT PASS
                3400 *
                3410 *        END OF FFT ROUTINE
                3420 *
4512- 60        3430 DONE     RTS
4513- 00 00     3440          .DA 0            ROOM FOR JUMP EXIT
                3450 *
                3460 *
                3470 *        -----------------
                3480 *         SCALING ROUTINE
                3490 *        -----------------
                3500 *
                3510 *        CHECKS ALL DATA TO ANTICIPATE POSSIBLE
                3520 *        OVERFLOW IN THE NEXT PASS.
                3530 *        IF NECESSARY DATA IS SCALED DOWN BY 2
                3540 *
4515- A2 00     3550 SCALE    LDX #0           INIT LOOP COUNTER
4517- BD 00 41  3560 SCL1     LDA REAL,X       GET VALUE
451A- 10 06     3570          BPL SCL11
451C- C9 C0     3580          CMP #$C0
451E- 90 19     3590          BCC SCL2         OV IF <$C0
4520- B0 04     3600          BCS SCL12
4522- C9 41     3610 SCL11    CMP #$41
4524- B0 13     3620          BCS SCL2         OV IF >$40
4526- BD 00 42  3630 SCL12    LDA IMAG,X       DO IMAG
4529- 10 06     3640          BPL SCL13
452B- C9 C0     3650          CMP #$C0
452D- 90 0A     3660          BCC SCL2         OV IF <$C0
452F- B0 04     3670          BCS SCL14
4531- C9 41     3680 SCL13    CMP #$41
4533- B0 04     3690          BCS SCL2         OV IF >$40
4535- E8        3700 SCL14    INX              INCREM LOOP COUNTER
4536- D0 DF     3710          BNE SCL1         SEE IF DONE
4538- 60        3720          RTS              NO OVERFLOW
                3730 *
                3740 *        SCALE EVERYTHING DOWN
                3750 *
4539- E6 0B     3760 SCL2     INC SCLFCT       INCREM SCALE FACTOR
453B- A2 00     3770          LDX #0           INIT LOOP COUNTER
453D- 18        3780 SCL3     CLC
453E- BD 00 41  3790          LDA REAL,X       GET VALUE
4541- 10 01     3800          BPL SCL4         PUT SIGN BIT IN CARRY
4543- 38        3810          SEC
4544- 6A        3820 SCL4     ROR              DIVIDE BY 2 WITH SIGN
4545- 9D 00 41  3830          STA REAL,X
4548- 18        3840          CLC
4549- BD 00 42  3850          LDA IMAG,X       DO SAME FOR IMAG
454C- 10 01     3860          BPL SCL5         PUT SIGN BIT IN CARRY
454E- 38        3870          SEC
454F- 6A        3880 SCL5     ROR              DIVIDE BY 2
4550- 9D 00 42  3890          STA IMAG,X
4553- E8        3900          INX              INCREM LOOP COUNTER
4554- D0 E7     3910          BNE SCL3         SEE IF DONE
4556- 60        3920          RTS
                3930 *
                3940 *
                3950 *
                3960 *
```


contd.

```
                    3970 *      ------------------------
                    3980 *       MULTIPLICATION ROUTINE
                    3990 *      ------------------------
                    4000 *
                    4010 *      THIS ROUTINE ASSUMES MCD IS AN 8 BIT
                    4020 *      SIGNED BINARY NUMBER.
                    4030 *      IT PERFORMS A SIGNED MULTIPLICATION WITH AN
                    4040 *      8 BIT 2'S COMPLEMENT NUMBER IN MPR.
                    4050 *      THE RESULT IS A 2'S COMPLEMENT NUMBER
                    4060 *      ROUNDED TO 8 BITS.
                    4070 *
                    4080 *      ALL INPUT VALUES DESTROYED (LOC 6 - A)
                    4090 *      Y-REG DESTROYED
                    4100 *
                    4110 *
4557- 85 08         4120 MPY    STA MPR          SAVE ACCUM
4559- A9 00         4130        LDA #0
455B- 85 07         4140        STA MCD+1        ZERO MCD+1
455D- 85 09         4150        STA RES          AND RESULT
455F- 85 0A         4160        STA RES+1
4561- A0 07         4170        LDY #7           LOOP COUNTER
4563- A5 08         4180        LDA MPR          COMPUTE SIGN OF RESULT
4565- 45 06         4190        EOR MCD
4567- 18            4200        CLC
4568- 08            4210        PHP              SAVE RES SIGN ON STACK
4569- 06 06         4220        ASL MCD          REMOVE SIGN FROM MCD
456B- A5 08         4230        LDA MPR          MAKE MPR POSITIVE
456D- 10 06         4240        BPL NEXT         MPR SIGN IS ALREADY +
456F- 49 FF         4250        EOR #$FF
4571- 69 01         4260        ADC #1
4573- 85 08         4270        STA MPR
4575- 46 08         4280 NEXT   LSR MPR          GET LSB OF MPR
4577- 90 0D         4290        BCC SHIFT        IF 0 DO NOT ADD
4579- 18            4300        CLC
457A- A5 06         4310        LDA MCD          GET MCD
457C- 65 09         4320        ADC RES          ADD TO RES
457E- 85 09         4330        STA RES
4580- A5 07         4340        LDA MCD+1        ALSO HIGH BYTES
4582- 65 0A         4350        ADC RES+1
4584- 85 0A         4360        STA RES+1
                    4370 *
4586- 06 06         4380 SHIFT  ASL MCD          SHIFT MCD
4588- 26 07         4390        ROL MCD+1
458A- 88            4400        DEY              DECREM LOOP COUNTER
458B- D0 E8         4410        BNE NEXT         SEE IF DONE
458D- A9 00         4420        LDA #0
458F- 06 09         4430        ASL RES          ROUND OFF RES TO 8 BITS
4591- 65 0A         4440        ADC RES+1
4593- 28            4450        PLP              GET RESULT SIGN
4594- 10 04         4460        BPL PLUS
4596- 49 FF         4470        EOR #$FF         CHANGE SIGN OF RES
4598- 69 01         4480        ADC #1
459A- 60            4490 PLUS   RTS              RESULT RETURNS IN ACC
                    4500 *
                    4510 *
                    4520 *
                    4530        .EN
```


contd.

SYMBOL TABLE

INPUT	4000	REAL	4100	IMAG	4200
SINE	4300	MCD	0006	MPR	0008
RES	0009	SCLFCT	000B	CELNUM	000C
DELTA	000D	PAIRNM	000E	PAIRCT	000F
CELDIS	0010	CELCT	0011	PTRM	0012
PTRN	0013	SINPT	0014	SINA	0015
COSA	0016	TREAL	0017	TIMAG	0018
TM	0019	BITREV	4404	BRV1	4408
BRV2	4429	PASS1	4431	NPASS	4459
NCELL	4466	NC1	446E	NXTCEL	44F7
NP1	4505	DONE	4512	SCALE	451C
SCL1	4517	SCL11	4522	SCL12	4526
SCL13	4531	SCL14	4535	SCL2	4539
SCL3	453D	SCL4	4544	SCL5	454F
MPY	4557	NEXT	4575	SHIFT	4586
PLUS	459A				

:

GREENAPPLES

BY Andy Rose

AND SOFTWARE THAT MAKES YOU SMILE

Software Reviews

Wilderness Campaign
from Synergystic Software

This program is an adventure game situated on a hi-res board. The board has mountains, swamps, deserts, jungles, plains and badlands. The location of these terrains are set. The board also contains randomly placed temples, tombs, castles, towns and ruins. The primary object of the game is to build up an army of 50 to 70 men who are fully equipped, and to take over the evil Necromancer's castle. The secondary objective is to get rich and become well equipped. You do this by fighting monsters, getting gold and buying armor and weapons. You can also gain magical items (one powerful item is needed to attack the Necromancer's castle). You can also get caught in avalanches, sandstorms, pit traps, poison vapor, etc. You need certain things to survive and travel, i.e., food, torches, 15' planks, canteens, machetes, etc. Everything in this games is based around gold quadroons - 1 quadroon to 100 coppers, 1 gold to 10 silvers.

The only flaw that I found in the whole game is that the number of mules and men you can have to carry weight is limited by line 50 in the program. (Line 50 states that men can carry 200 and mules can carry 1000 lbs. If this number of men and mules goes over 32767, then an error occurs.) One especially nice thing about the program is that if you press 'reset' instead of 'return' you can restart the program by a GOTO 10 without losing your place in the game. Overall, I think that this game is top of the line and you won't become bored (unless you cheat!).

Bill Budge's Space Album

This disk is one of the best that I have ever seen. It contains four programs, all of which are top notch.

a) Asteroids

This program is based on the Asteroids game in the arcades. It has a triangular ship in the center that shoots passing asteroids. To fire you press the Paddle 0 button and to aim you turn Paddle 0. The game keeps a high-score, and has four levels of play.

b) Death Star

In this game the object is to fly through a trench, blasting T.I.E. fighters and laser turrets, and drop a bomb down the reactor vent. This game is very challenging and takes a long time to master. While you are flying down the trench, you must dodge enemy fire, avoid hitting the trench walls, and shoot at other ships, or you will be destroyed. This game has only one level of play (hard) and keeps a high-score.

c) Tailgunner

You are a tailgunner in the back of a cargoship trying to make its galactic run without being destroyed by space pirates. You have to destroy the hostile crafts before they pass you or else you have to jump into hyperspace. You better shoot good, because the ship can only stand ten jumps! This game has the best graphics of all four games, has two levels of play, and keeps a high-score.

d) Solar Shootout

In this two person game, you fly around the sun trying to blow the other person to the cosmos. You go faster or slower by going closer to or farther from the sun. This game involves strategy and action all in one.

All of these games use paddles, have sound, and are top quality. WARNING! If you play these games for a while, you'll notice that your finger starts to hurt. Wear protective clothing!!

PROGRAMMER OF THE MONTH
(A new column)

Did you just buy some software that you really liked and would like to recognize the author. Now you can!!. Just write down your first, second and third choices for Programmer of the Month on a postcard or letter and send to:

P.O.T.M.
5455 Wingsborne Ct.
Columbia, MD 21045

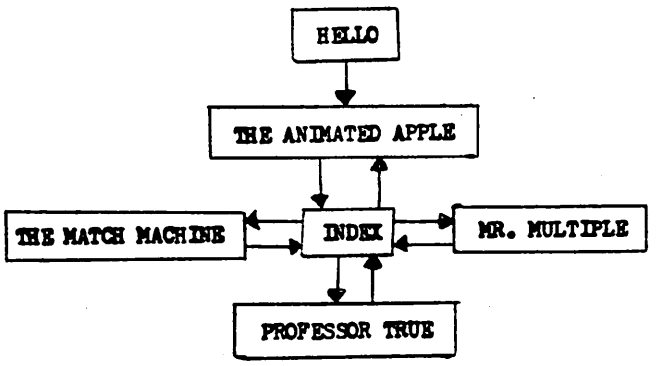
If the program that you like doesn't have an author, then write the name of the company.

THE SHELL GAMES

BY Bruce Tognazzini
A REVIEW BY Tom Lucas

Why is the peanut not a nut? When did the first nuclear reaction go critical? Who is Lamont Cranston? What do you call a group of apes? The answers to these questions and more can be found in THE SHELL GAMES.

THE SHELL GAMES turn your APPLE into a teaching machine. When you buy this program, you get a user manual and six programs on a disk, arranged to call one another in the following manner:



When the disk is booted, the HELLO program automatically runs THE ANIMATED APPLE. This is a Lo-Res cartoon which shows an apple tree bud, blossom, and grow an apple. The apple ripens into the APPLE logo, and falls to the ground, welcoming you to THE SHELL GAMES.

The INDEX program is run next. It is a menu to the three educational programs on this disk: THE MATCH MACHINE (a matching quiz), PROFESSOR TRUE (a true/false quiz), and MR. MULTIPLE (a multiple choice quiz).

The three educational programs take the form of problem shells in which the user can easily enter his own data. Each program shares the same anatomy which includes a PROBLEM ENTRY EDITOR. This is used to add, delete, or edit problems. All of the scoring, rewards, editing and other mechanics of presenting the problems are programmed around the problem shells.

THE SHELL GAMES

THE MATCH MACHINE comes with 15 separate match problems. Each problem lists a column of one to ten words and scrambles a list of match words. The quiz allows for multiple right answers and words that have no match. Some of the matching quizzes that are included are Sports Superstars, The Inventors, and Animal Groups.

PROFESSOR TRUE comes as a series of true/false problems on American History. After each problem there is an explanation of the answer and a little more background information. PROFESSOR TRUE is designed for very young players. There are lots of lights, sounds and rewards for making a correct selection. By the very nature of a true/false quiz there are only two possible answers, so the very young student can easily play PROFESSOR TRUE.

MR. MULTIPLE comes as a series of multiple choice problems on little known facts and folklore. Like PROFESSOR TRUE, after you select the correct answer, there is an explanation and more background information. After playing MR. MULTIPLE you'll know what the fastest animal in the world is and the name of the longest running TV series.

DESIGN PHILOSOPHY

Although THE SHELL GAMES are structured as a series of problems, they are not really meant to quiz but to teach. The "backward" structure of question then information is designed to first spark your interest in the topic, and then teach you about the subject. There is no penalty for missing questions, only rewards for a correct response. At the end of the game you are encouraged to go back and retake missed questions.

THE PROBLEM ENTRY EDITOR

The PROBLEM ENTRY EDITOR is used to build new quizzes. By LOADing one of the shell games and typing "RUN2" you will enter the PROBLEM ENTRY EDITOR. From here, you can easily change all or part of any problem or add new problems. The editor was designed so that anyone with about ten hours experience with the APPLE computer can use it. The teacher or the student can use the PROBLEM ENTRY EDITOR. A very sophicticated lesson does not require a sophisticated programmer. If the student constructs his own lesson, he will save time for the teacher, learn about computers, and automatically learn the quiz material.

Another feature of each shell game is a series of switches. By setting or unsetting these switches, the user can customize the shell game to his own particular need. For example, you can adjust the amount of encouragement or amount of reward by changing the value of two switches. You can also force the student to retake all missed questions by setting one switch. The program will check these switches and even change the instructions to coincide with the setting of the switches.

CONCLUSION

THE SHELL GAMES are a very good way to put your APPLE to work as a teaching machine. I found the instruction manual very complete and the programs very easy to use. Designing a well thought-out quiz is a difficult job, but with the shell games programs, it is easy to enter and edit your quizzes.

DOS 3.3 AND THE LANGUAGE SYSTEM

by Howard Lefkowitz

After several phone calls to Apple, Inc., I believe I have resolved the question raised in the August 1980 Apple Pi NEWSIG Notes column about the compatability of DOS 3.3 with the PASCAL System.

1. PASCAL uses 16 sector disks and the Language System provides the P5A and the P6A PROMS needed. These PROMS are identical to those provided with a DOS 3.3 package.

2. The disk named "BASICS" provided with the Language System is used to boot DOS 3.2.1 and also boot the Basic language which is not in ROM in your computer (Integer or Applesoft).

3. The disk named "BASICS" which you receive with the DOS 3.3 package is identical to the one in the Language System.

4. If you do not have the Language System the "BASICS" disk simply boots DOS 3.2.1 and allows you to run disks of this type.

5. The DOS 3.3 disk, supplied with this package, provides the 3.3 system. If you have a Language System this disk boots 3.3 and ALSO boots the Basic language which is not in ROM into your language card.

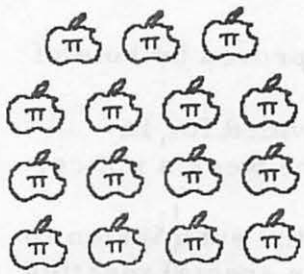
To summarize - If you have a Language System:

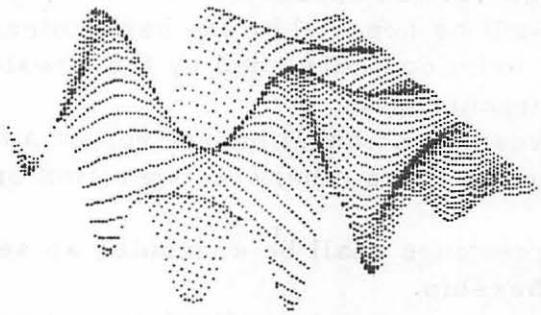
For DOS 3.2.1 Boot the "BASICS" disk.
For DOS 3.3 Boot the "DOS 3.3" disk.
For PASCAL Follow the instructions.

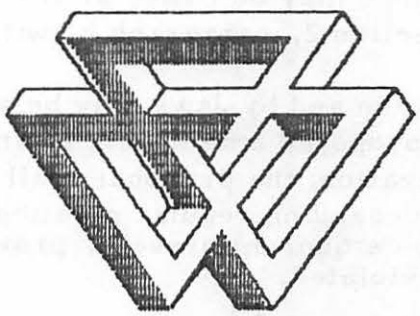
Reprinted from the Apple Cart

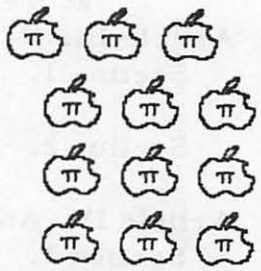
I know that there are a lot of you out there who, like me, have the Pascal Language System and who, like me, would like to use lower case. Well, here is one solution provided free courtesy of Peripherals Unlimited:

```
(* Program to modify the Bios Modules*)
(* To work with Dan Paymar's lower case adapter*)
(* Developed and tested by Craig Vaughan*)
(* Copyright waived*)
PROGRAM LC UPDATE;
VAR
  BLK: PACKED ARRAY [0..511] OF 0..255; (*
    BLOCK TO HOLD BIOS CODE*)
  BLT, (* # OF BLKS TRANSFERRED*)
  BLN: INTEGER; (* BLOCK # TO READ <4> *)
  S : FILE; (* FILE ID *)
  SRC : STRING; (* FILENAME*)
BEGIN
  SRC : = 'SYSTEM.APPLE'; (* FILE CONTAINING
    BIOS *)
  RESET (S,SRC); (* OPEN IT*)
  BLN : = 4; (*READ BLOCK #4 *)
  BLT : = BLOCKREAD(S,BLK, 1,BLN); (*READ IT*)
  BLK [232] : = 234; (* NOP CASE CONVERSION*)
  BLK [233] : = 234; (* " " " *)
  BLK [235] : = 127; (* CHANGE CHARACTER
    MASK*)
  BLT : = BLOCKWRITE (S,BLK,],BLN); (*WRITE IT
    BACK OUT*)
  CLOSE (S,LOCK); (*CLOSE AND LOCK IT*)
  END. (*ALL DONE*)
```









CONSTITUTION AND BY-LAWS
of
WASHINGTON APPLE PI

Preamble: We the members of Washington Apple Pi do organize for mutual learning and education.

CONSTITUTION:

Article I. Name.

Section 1. We shall be known as Washington Apple Pi.

Article II. Officers; Duties of Officers.

Section 1. The Executive Board of Washington Apple Pi shall consist of a President, Vice-president, Secretary, Treasurer and three at-large members.

Section 2. President shall:

a. Preside at meetings.
b. Appoint an Editor and/or Librarian.
c. Call extra meetings, appoint committees not otherwise provided for, and fill committee and protem vacancies with the consent of the Executive Board
d. Oversee the efforts of the appointments.
e. Perform such other duties as the office may require.
f. Perform the duties of the Vice-president in the latter's absence.

Section 3. Vice-president shall:

a. Perform the duties of the President in the absence of that officer.
b. Perform the function of Administrative Assistant.
c. Oversee and coordinate the efforts of all committees.

Section 4. Secretary shall:

a. Keep the minutes of all business meetings.
b. Maintain an up-to-date roster of members.
c. Maintain the correspondence.

Section 5. Treasurer shall:

a. Keep an accurate account of all financial transactions of the organization.
b. Be responsible for all funds of the organization, receive and disburse all money, subject to the will of the other elected officers.
c. Disbursements over $25 shall be made only by check. Disbursements over $5 shall be supported by voucher or receipt. At such time that the funds of the organization exceed $5, a checking account shall be set up and no check shall be honored by the bank unless signed by the Treasurer, such signature to be countersigned by the President or one other elected officer of the organization.
d. Prepare and present an annual budget report and forecast. This shall be at the June meeting or as otherwise specified or directed by the President.

Article III. Meetings.

Section 1. Regular meetings shall be scheduled as set and approved by vote of the membership.

Section 2. Special meetings may be called at any time as provided for in Article II, Section 2, paragraph c, with at least one week's notice.

Article IV. Amendments.

Section 1. This constitution and by-laws may be amended by the submission in writing of a proposed amendment at any regular or special meeting of the organization; the proposal shall be received and acted upon at the next succeeding regular meeting of the organization at which there shall be a quorum present, providing Article IV, Section 3, shall not be violated.

Section 2. Amendments may be acted upon at the announced meeting or by mail ballot at the choice of the President. A two-thirds majority of the members present at the meeting or responding by mail ballot shall be required to adopt or reject the amendment.

Section 3. Written notice of the meeting containing the proposed amendment shall be given two weeks prior to the meeting at which the amendment is voted upon.

BY-LAWS:

Article I. Membership.

Section 1. Membership shall be open to all without regard to race, creed, color, sex, age, national origin, or sexual preference.

Section 2. Dues shall be set and approved by vote of the membership. Membership in this organization by one member of a household may include all members of that household as one member.

Section 3. The dues year shall run from June 1 to May 31. Dues paid after March 1 shall be for the following year.

Article II. Quorum.

Section 1. A quorum exists in the presence of the lesser of 15 members or 50 percent of the roster of members.

Article III. Election of Officers.

Section 1. The offices of the Executive Board shall be filled by election at the first regular meeting in May. They shall take office on June 1.

Section 2. Nominations for officers of the organization shall be submitted at the first regular meeting in April. The nominations will be received from the floor, or by mail prior to the April meeting.

Section 3. No member is eligible to hold more than one elective office within this organization at one time.

Section 4. All elections shall be by secret ballot; a simple majority of the votes cast is required for election.

Section 5. Special elections in the event of office vacancies may be held at any time by acclamation from the members present.

Article IV. General Provisions.

Section 1. Property or equipment purchased by this organization with the organizations's funds shall not be considered the property of any individual member or group of members, but shall be known as Club property.

a. In the event of Washington Apple Pi disbanding, Club property and funds shall be disbursed in a manner agreeable to two-thirds of the remaining members.

Section 2. Acquistion of any property, real, personal or interest therein, in excess of $100 valuation shall receive prior approval by the membership.

Section 3. The fiscal year shall be from June 1 to May 31. All budgeting and financial accounting shall be rendered on this basis.

Article V. Activation of the Constitution and By-Laws.

Section 1. This constitution and by-laws shall be made to become in effect upon the endorsement of at least 10 charter members.

Section 2. Upon activation, offices shall be filled under provisions of By-Laws Article III, Section 5.

YOUR AD HERE !

RATES		
	$30	full
	$ 15	half
	$ 10	quarter
	$ 6	eighth

(line copy only - no half-tones or colors)

WASHINGTON APPLE PI
MAIL ORDER FORM

Washington Apple Pi now has a program library, and disks are available for purchase by anyone. The price to members is $5.00 per disk and $8.00 to non-members. These disks are chock full of exceptional programs - the utilities are especially useful. The games are some of the best - not just simple and uninteresting ones. You may pick them up at any meeting or have them mailed for $2.00 per disk additional. They will come in a protective foam diskette mailer.

Also available for purchase by members at a discount price is the new APPLE II REFERENCE MANUAL (replaces the Red Reference Manual). The price of this manual is $17.00. You may pick it up at a meeting or have it mailed to you at no extra charge.

Amount

1. New APPLE II REFERENCE MANUAL - $17.00 each ------

2. PROGRAM DISKETTES
   Members: $5.00 picked up at meeting
   $7.00 mailed to you...

   Non-members: $8.00 per disk picked up at meeting
   $10.00 mailed to you...

Volume 1	Utilities I	( )	Volume 22	Utilities VI	( )
Volume 2	Utilities II	( )	Volume 23	Games VIII	( )
Volume 3	Games I	( )	Volume 24	Games IX	( )
Volume 4	Games II	( )	Volume 28	Planetfinder	( )
Volume 5	Games III	( )	Volume 180	Dungeon Designer	( )
Volume 6	Games IV	( )	Volume 181	Beginner's Cave	( )
Volume 7	Games V	( )	*Volume 182	Lair of Minotaur	( )
Volume 8	Utilities III	( )	*Volume 183	Cave of the Mind	( )
Volume 9	Educational I	( )	*Volume 184	Zyphur Riverventure	( )
Volume 10	Math/Science	( )	*Volume 185	Castle of Doom	( )
Volume 11	Graphics I	( )	*Volume 186	Death Star	( )
Volume 12	Games VI	( )	*Volume 187	Devil's Tomb	( )
Volume 13	Games	( )			
Volume 14	IAC Utilities IV	( )			
Volume 15	Games VII	( )			
Volume 16	Utilities V	( )			
Volume 17	Graphics II	( )			
Volume 18	Educational II	( )			
Volume 19	Commumications	( )			
Volume 20	Music	( )			
Volume 21	Apple Orchard	( )			

*Vol. 181 required with these disks.

TOTAL ORDER = $ ------

Check here if you want these shipped---

NAME ----------

ADDRESS ----------

CITY, STATE, ZIP ----------

TELEPHONE ----------

Membership No.(1st three digits after WAP on mailing label) --------

Make checks payable to "Washington Apple Pi"

Send order to: Washington Apple Pi- ATTN: Librarian
PO Box 34511
Washington, DC 20034