CURRENT NOTES

The Newsletter for ATARI Users of D. C. and Northern Virginia

Volume 2, Number 10
October, 1982

DC CURRENTS

Oct. 19: Great Moments in Show and Tell

Demonstrations will be the order of the evening at the DC meeting on Tuesday, October 19. Craig Smith will bring his system with two added attractions which he has installed in it, the Axlon 128K Ramdisk and the BIT3 80-column display board. Craig had the BIT3 board at an earlier meeting, but its capabilities were restricted by the monitor which was available. Craig will bring along a green-screen monochrome monitor which lets the 80-column display show its true colors (!). The Ramdisk, a 128K memory board which looks to the Atari like a disk drive, or else can be used as extra memory with bank selection, has received a lot of publicity. This will be an opportunity to see exactly how it works and how it can be used.

At the same meeting, Frank Huband will demonstrate the VoiceBox, the newest speech synthesizer device for the Atari, and the "math chip" which is alleged to speed up computational operations significantly. There has been quite a bit of commentary, not to mention advertising, about these new products in the computer magazines in the last few months. Here's a chance to see them in action.

The meeting will be in Room 1240 of the National Science Foundation offices at 1800 G Street, Northwest, in Washington. Closest Metro stop is Farragut West (take the 18th Street exit). Parking is available in the building, for a price. The parking entrance is on 18th St., between F & G. The meeting will begin about 5:30 PM.

One of the favorite activities in Washington is demonstrations near the White House. Here's your chance to come to one. See you there.

NOVATARI NOTES

Printers Revisited, October 10

Did you know that Jack Gutenberg set up his printing shop at just the same time as Chris Columbus organized his sailing excursion business? If you did, you're wrong, because actually it was about 50 years earlier, but no matter--it was the same century, and as we all know, years were closer together back then. In any event, Novatari will celebrate the Columbus day weekend with an extravagant display of printing power for our favorite computer. A variety of familiar and exotic machines will be displayed, demonstrated, and discussed at our meeting on Sunday, October 10. Steve Steinberg will bring the popular Epson MX-80/FT, with GRAFTRAX +. Atari's own 825 will be represented by that celebrity of printers, the very printer which produces Current Notes month after month (that awesome fact ought to be enough by itself to get you to the meeting). Rene Hertz enters the heavyweight division with the Anderson-Jacobson 100-column daisy-wheel printing terminal, and adds color to the occasion with the new Prism color printer. There may be some other surprises as well. If you're interested in the current state of the printer's art, microcomputer division, you should find this meeting worth your while.

We will return to the Warner Amex Cable TV studio for our October meeting, probably for the last time. Arrangements are under way for a new meeting site, but they haven't been completed yet. Be sure to watch next month's Current Notes carefully for news about this. The Warner Amex Cable TV studio is on Victory Drive in Herndon, which is a little shopping center/industrial complex fronting on Spring Street, which runs from Herndon to Reston parallel to and just north of the Dulles Access Road. People will begin to gather a little after 5:00 PM for casual conversation and commerce. The organized festivities will begin at 7:00 PM, and may last an hour to an hour and a half, depending on how interested everybody is.

(over)

Novatari Notes (cont'd.)

The Old Order Changes: with an appreciation of Tom Bartelt

Tom Bartelt of Reston was the person who took the initiative to start our group a year and a half ago, and has served as our host, treasurer, and emcee ever since. Now some of the patterns of Tom's life are changing, and he has asked to be relieved of some of his ongoing responsibilities to the group. It's a measure of Tom's contribution to the group that it is taking three people to replace him. At the September meeting, Curt Sandler agreed to serve as Treasurer, and Tim Kilby agreed to serve as Chief of Correspondence (your editor just made up that title)--Tim will be the contact person on the lists of user groups at Atari, Inc., which seem to have fairly wide circulation among other groups, vendors, and such like, all of which generate a certain volume of mail to deal with. A small committee was formed to locate a new meeting place, and probably one or another member of that committee will end up being responsible for meeting arrangements.

All of us who enjoy and benefit from Novatari owe a round of grateful applause to Tom Bartelt.

Fourth-Sunday Meetings Put on Hold

Up through last May, it was Novatari's practice to meet twice a month, on the second and fourth Sundays of each month. Summertime, the livin' is easy, as you know, so we cut back to a once-a-month schedule, meeting just on the second Sunday. Now that we're in a period of uncertainty about exactly where we're going to be able to meet, and whether our next host may not be so generous as not to charge us for the space, we really can't resume the twice-monthly schedule yet. After things settle down and we know what's what, we'll determine the wishes of the membership as to the frequency of meetings, and take it from there. Meanwhile, some of you may be interested in taking up the slack by organizing special interest groups--games, education, assembly language programming are some of the topics that have been suggested. All that's needed is a little leadership. The pages of Current Notes are open to any member who would like to get such a group started.

For the time being, until further notice, Novatari will meet just on the second Sunday of every month.

CLASSIFIED

FOR SALE: 3 Atari disk drives, latest revision, including PERCOM data separator boards, $300 to $350. Epson MX-80/FT with GRAFTRAX, just totally overhauled, with 1-year factory warranty, $520. Call Rene Hertz, 860-2046.

CAR POOL: Room for 2 in car pool to Atari Assembly Language course in Silver Spring, Maryland, October 16, 23, and 30. Call Rene Hertz, 860-2046.

CURRENT NOTES is the monthly newsletter sent to members of the ATARI Club of downtown D. C. and Novatari (the Northern Virginia ATARI Users' Group). Both of these organizations are independent groups for computer users, and neither group is affiliated in any way with ATARI, Inc.

The Editor of CURRENT NOTES is Paul Chapin, 2159 Golf Course Dr., Reston, Va. 22091, telephone (home) 476-5950, (office) 357-7696. News items, short articles, original programs, classified ads, and any other material of interest to the membership are eagerly solicited.

Membership dues for both groups are $15.00 a year, which includes subscription to CURRENT NOTES. Dues are payable at the beginning of each calendar year. Dues for new members joining during the year are reduced $1.00 for each month which has passed since the first of the year. Dues may be paid at any meeting, or be sent to the editor. Persons living outside the metropolitan Washington D.C. area may subscribe to CURRENT NOTES for $12.00 per year.

Advertising policy: classified ads are free to members. Commercial advertising rates are $10.00 for a page, $5.00 for a half page (no other fraction available). Advertising for any month's issue must reach the editor by the 20th of the preceding month. Advertising must be in the form of xerox-ready copy, on an 8 1/2 x 11 sheet for a full page or an 8 1/2 x 5 1/2 sheet for a half page. Full pages are reduced to 7 x 8 1/2, half pages to 7 x 4 1/4.

SECRET WRITING
by Jim Bumpas
A.C.E. Newsletter
Eugene, Oregon

Key-6 Hand Cipher "Secret writing" is as old as writing itself. In fact, at first ALL writing was secret, as only the "initiates" could decipher the meaning. Even today, all writing remains secret to more than half the people in the world who can read no writing. As more and more people began to learn to read, the need for secret writing, or "ciphers", became important.

Ciphers began to become very important to diplomacy as kings and republics began to conduct international relations. By the 17th century, ciphers had also become a hobby among some literate individuals. Military ciphers were first extensively used operationally in the Civil War. The Union read most of the Confederate ciphers; the Confederates read very few Union ciphers. Edgar Allen Poe is probably the most famous cipher hobbyist and ciphers play a part in several of his novels.

Hand ciphers have been a hobby of mine for years. I've invented more than I've solved, except for the little short ones in magazines. A "hand" cipher is one which can be reasonably used with pencil and paper to encipher and decipher (with the key) messages. By "reasonably" I mean one must be able to encipher and decipher messages at the rate of about 10 words per minute.

Each such cipher I've invented, I thought to myself: THIS one is impossible to break! Needless to say, they wre all broken. Some of my "better ones even more easily than some others.

A year or so ago, I invented a new one. This one has not been broken after nearly a year of trying by several other cipher hobbyists. This cipher almost completely avoids any normal letter frequency count. Theoretically, each letter could appear as often as any other in this cipher. In practice, however, it seems there still is repetition of some significant letter groups which make it easier to break than had those repetitions not occurred. This cipher is even superior to the WW2 German Enigma in the one respect (at least) that a letter may be enciphered as itself.

Well, when I recently bought an Atari 800, my first project has been to reduce the cipher to arithmetical expression and program the machine to do the enciphering and deciphering. Linda Bumpas is an experienced programmer, making the project possible. I'm sharing the program Linda and I devised to generate this cipher with the readers here because the program itself might be of interest to you, even if you are not presently a cipher hobbyist.

As you can see from the "Key-6 Hand Cipher" program, the cipher is based upon 2 shifting keys of letters, which serve to produce what may be called a "polyalphabetic substitution cipher." When the idea came to me, it appeared that theoretically, a message could be enciphered into a number of keys equal to 676 times the number of letters in the message. In practice, and for reasons I don't entirely understand, this doesn't appear to be the case. The repetitions I mentioned earlier are probably an influence in this result. It seems the influence of those letters which normally produce high frequency counts in the usual transposition cipher (of which this is one) causes repetitions which constrict what I thought was the theoretical number of keys possible. At any rate, while I'm not sure how many keys are made available by this cipher, it does appear to be a very large number, which makes breaking it relatively difficult. I don't use the word "impossible" any more as I've seen too many of mine broken. The Allies even broke many of the German Enigma ciphers in WW2.

This program also includes instructions to read output into a tape cassette, and to read input from a tape cassette. You may use the program to send enciphered messages to your friends by tape cassette. Only someone with the "key" can translate the ciphertext message into "plaintext" so it can be read.

I invite responses by readers to any experiences you may have with this cipher, its program, or any mathematical tests you may devise to run upon it. I'll share with you further, too, anything more I learn from it.

-Jim Bumpas, 1405 West 26th Avenue, Eugene, OR 97405

```
10 REM KEY 6 CIPHER BY JIM & LINDA BUMPAS,
JULY, 1981
12 REM FROM A.C.E. NEWSLETTER (EUGENE,
OREGON)
30 CLR :DIM Y$(1)
40 DIM IN$(800):DIM OUT$(800)
50 DIM T1$(26):DIM T2$(26)
60 T1$="ETIOANSHURMDJCVBFLPGWKYXQZ"
70 T2$="DQMYJXCZVWBKGRFUPHELSTNIAO"
80 GRAPHICS 2+16:POSITION 5,2:? #6;"KEY-6
HAND":POSITION 7,4:? #6;"CIPHER":POSITION
9,6:? #6;"by"
82 POSITION 0,8:? #6;"JIM AND LINDA BUMPAS"
90 FOR X=1 TO 30:SETCOLOR 4,X/10,2
92 SOUND 2,12,12,12
94 R=INT(RND(1)*40)+20
96 FOR S=1 TO R:NEXT S
98 SOUND 2,0,0,0:NEXT X
```

(over)

```
 GRAPHICS 2+16:SETCOLOR 4,5,4
 POSITION 3,2:? #6;"I AM A CIPHER":POSITION
 ,3:? #6;"MACHINE. I ENCIPHER":POSITION 1,4
 POSITION 1,4:? #6;"PLAINTEXT MESSAGES"
 POSITION 4,5:? #6;"AND DECIPHER"
 POSITION 0,6:? #6;"CIPHERTEXT MESSAGES."
104 FOR X=1 TO 2000:NEXT X
108 POKE 82,4
109 OPEN #2,4,0,"K:":OPEN #3,8,0,"S:"
 10 ? "IS THE MESSAGE ON TAPE?(Y/N)"
 12 INPUT Y$:IF Y$="Y" THEN GOTO 1000
 30 ?
 40 ? "DO YOU WISH TO (E)NCIPHER A PLAINTEXT
 R (D)ECIPHER A CIPHERTEXT MESSAGE?":?
 50 ?
 60 ? "TYPE 'E' OR 'D' NOW.":?
 70 INPUT Y$:IF Y$="D" THEN 420
180 ? "WHEN YOU'RE READY, TYPE THE
PLAINTEXT MESSAGE, PLACING AN '*' AT THE
END.":?
190 GOSUB 1100
200 GOSUB 640
205 POKE 559,0
210 TX=0:IOX=1
220 ?
230 IF IN$(IOX,IOX)="*" THEN 400
235 GOSUB 800
240 X1=1
250 IF IN$(IOX,IOX)=T1$(X1,X1) THEN 290
260 X1=X1+1
 70 IF X1>26 THEN 380
 80 GOTO 250
 90 IF X1+TX<=26 THEN 320
300 OUT$(IOX,IOX)=T2$(X1+TX-26,X1+TX-26)
310 GOTO 330
320 OUT$(IOX,IOX)=T2$(X1+TX,X1+TX)
330 TX=X1+TX
340 IF TX>26 THEN TX=TX-26
350 IOX=IOX+1
360 IF IOX<=800 THEN 230
370 GOTO 400
380 OUT$(IOX,IOX)=IN$(IOX,IOX)
390 GOTO 350
400 GOSUB 760
402 ? "DO YOU HAVE ANOTHER MESSAGE?  TYPE
 Y' OR 'N'":?
 04 INPUT Y$:IF Y$="Y" THEN 109
 06 ? "DO YOU WISH TO SAVE THIS MESSAGE ON
 APE? (Y/N)"
 08 INPUT Y$:IF Y$="Y" THEN GOSUB 900
 10 END
 20 ? "WHEN YOU ARE READY, TYPE THE
 IPHERTEXT MESSAGE, PLACING AN '*' AT THE
END."
421 ? :GOSUB 1100:?
440 GOSUB 640
445 POKE 559,0
450 ?
460 TX=0:IOX=1:LX=0
470 IF IN$(IOX,IOX)="*" THEN 620
475 GOSUB 800
480 X1=1
490 IF IN$(IOX,IOX)=T2$(X1,X1) THEN 530
500 X1=X1+1
510 IF X1>26 THEN 600
520 GOTO 490
530 TX=X1-LX
540 IF TX<=0 THEN TX=TX+26
550 LX=X1
560 OUT$(IOX,IOX)=T1$(TX,TX)
570 IOX=IOX+1
580 IF IOX>800 THEN 620
590 GOTO 470
600 OUT$(IOX,IOX)=IN$(IOX,IOX)
610 GOTO 570
620 GOSUB 760
630 GOTO 402
640 REM
650 X=1
660 GET #2,A
670 PUT #3,A
680 IF A=126 THEN 740
690 IN$(X)=CHR$(A)
700 IF IN$(X)="*" THEN 730
710 X=X+1
720 IF X<=800 THEN 660
730 RETURN
740 IN$(X-1)=" "
750 X=X-1:GOTO 660
760 CLOSE #2:CLOSE #3
765 POKE 559,34
769 GRAPHICS 0
770 ? :? IN$:?
772 FOR V=1 TO IOX-1:? OUT$(V,V);:NEXT V:? :?
780 RETURN
800 SOUND 2,12,12,12
810 R=INT(RND(1)*5)+5
820 FOR S=1 TO R:NEXT S
830 SOUND 2,0,0,0
840 RETURN
900 OPEN #1,8,0,"C:"
910 PRINT #1,OUT$
920 CLOSE #1:RETURN
1000 OPEN #1,4,0,"C:"
1010 INPUT #1,IN$
1015 Z=LEN(IN$)+1
1020 CLOSE #1:IN$(Z,Z)="*"
1030 GOTO 445
1100 ? "IF YOU MIS-TYPE, STRIKE 'DELETE BACK
S' FOR EACH CHARACTER IN ERROR.":? :?
1110 RETURN
```

TUTORIAL

HELPFUL HINTS FOR BASIC PROGRAMS

by David Bowen

There are three methods of keyboard input for BASIC programs. These methods are (1) INPUT statement, (2) GET statement and (3) PEEK(764). Together these provide a broad range of methods. Each is described below, with an example to illustrate its particular strengths. The differences between the methods occur in the following areas:

A. Is there any preparation required?

B. Does the computer stop while waiting for input?

C. Is there a question mark prompt and cursor displayed?

D. Does the computer respond to a keystroke with the "chirp" and/or symbol display?

E. Does the user need to press RETURN to end the input operation?

F. Does the programmer need to decode the resulting data?

With the above as an introduction, we now turn to describing the three methods.

1. INPUT statement. With this method, the program stops operation when the INPUT statement is reached (i.e. no more BASIC statements are executed until the input operation is finished).

The question mark prompt and cursor are displayed. There is a "chirp" for each keystroke, and the key symbol is displayed. RETURN must be pressed to input the data and have program execution continue. If you are writing a program for people not familiar with computers, this last requirement can be a problem, so include "PUSH RETURN" in the instructions you print. Three large advantages to the INPUT method are: (a) no decoding is required by the programmer (i.e. the data can be used without further processing), (b) more than one keystroke per variable is handled automatically, and (c) more than one variable per input operation can be handled automatically. One drawback to this method is that typing a non-numeric key when numeric input has been specified (e.g. by an INPUT X statement) will result in termination of the program with an error message, unless a TRAP statement is used.

EXAMPLE

```
100 TRAP 150
110 PRINT "TYPE PRINCIPLE,INTEREST
(COMMA BETWEEN)"
120 PRINT "THEN PUSH RETURN."
130 INPUT PRIN,INT
140 GOTO 170
150 PRINT "NON-NUMERIC INPUT TRY
AGAIN"
160 GOTO 110
170 ...Continue with program.
```

2. GET statement. With this method, the screen must be OPENed first (see example). The GET statement will cause the program to stop and wait for input, but without displaying the question mark prompt or the cursor. The keyboard will "chirp" with each keystroke, but the key symbol will not be displayed. If the prompt, cursor or key symbol display are wanted, they can be programmed as desired.

Program operation continues as soon as any key is pressed; use of the RETURN key is not required. Only one character is input per GET statement. Data is returned in ATASCII code, which can be easily interpreted (see example). The following example has two sections. The first section, lines 490-560, (a) illustrates alphabetic input and (b) prints all input, right or wrong, to the screen. The second section, lines 570 to 630, (a) accepts either numeric or alphabetic input, and (b) prints only appropriate input to the screen. Line 490, the OPEN statement, prepares the computer for bypassing the normal BASIC keyboard routines.

EXAMPLE:

```
490 OPEN #4,4,0,"K:"
500 PRINT "TYPE S (STOP) OR C (CONTINUE)"
510 GET #4,X:REM-INPUTS NEXT
KEYSTROKE AS X
520 PRINT CHR$(X):REM-PRINTS KEY
```

```
SYMBOL TO SCREEN
530 IF X=ASC("S") THEN END
540 IF X=ASC("C") THEN 570
550 PRINT "TYPE S OR C - TRY AGAIN"
560 GOTO 500
570 PRINT "TYPE MENU CHOICE 1-4,M FOR
MENU"
580 GET #4,X
590 IF X=ASC("M") THEN PRINT "M":GOTO
1000:REM-THIS SECTION WOULD PRINT
MENU
600 X=X-48:REM-CONVERTS ATASCII TO
NUMBER
610 IF X<1 OR X>4 THEN PRINT "TYPE 1-4
OR M - TRY AGAIN":GOTO 570
620 PRINT X:REM-ONLY PRINTS TO SCREEN
IF INPUT IS CORRECT
630 REM-ROUTE PROGRAM ACCORDING TO
MENU CHOICE
```

3. PEEK(764) method. By using the PEEK(764) method,the programmer can bypass all BASIC and operating system processing to tailor the most appropriate response. The PEEK at memory location 764 does not halt program operation; the contents of this location are simply transferred to the specified variable and operation continues. RETuRn is not used. The cursor, the "chirp" and the print to screen,if desired,must be programmed, but conversely,inappropriate keystrokes can be "masked out" or "disabled" by not programming responses to them. Alternatively,error messages can be printed for them.

In using this input method, a variable must be designated to receive the value stored in location 764. The value is transferred to, say, the variable X by the BASIC statement X=PEEK(764). Simple enough, but there are two problems that must be dealt with by the programmer.

The first problem is that location 764 is not reset after use by the Operating System. The contents of location 764 are simply the numerical code for the last key pressed. There are many cases where an unforseen keystroke can cause problems. For example, in executing a BASIC program, the last keystroke before execution begins is the RETURN to enter RUN. An early PEEK(764) would return the number 12, the code for RETURN. The solution for this problem is to POKE 764 with the CLEAR key code, 255, before the first PEEK and after every PEEK. See the second example below.

The second problem is that the code used is neither ATASCII nor the internal code used for screen printing, but a third code which is called the "keyboard code". This code is nowhere near as neat or rational as the other two. Numbers, letters and special symbols are out of order and interspersed. How then is this code to be interpreted?

There are two solutions. First you can crack the code yourself by running the first example program below and typing the keys you want the code for. The program will print the key symbol and the code. Running this program will also give you a good feel for the way the PEEK(764) method works. In your program, then, check for the keys you want to respond to by using IF statements to compare the variable to the codes, one by one. This is the best method to use if (a) you are only responding to a few keys, or if (b) you do not want to print the key symbol. This method is used twice, once for each reason, in the second example program below.

The second solution is to use the ROM table that converts from keyboard code to ATASCII. The table starts at location 65278 (hexadecimal $FEFE). If the Keyboard Code is X, then the ATASCII code is stored at location 65278+X. The use of this table is illustrated in the first example program below. Two warnings though:

A. The table location is not guaranteed to be the same in future versions of the Operating System. Presumably there is a "safe" vector somewhere that points to the table, but Atari hasn't told us (or at least me) where it is yet.

B. Lower case codes are returned if SHIFT is not pressed at the time. This is a somewhat different convention than that used by BASIC, so be careful. The first example program below shows how to convert the lower case ATASCII code to upper case by subtracting 32 from codes in the range for letters.

EXAMPLE 1 (illustrates the various codes):

continued on p.4

The Newsletter of the
Atari Computer Association of Orange County

June, 1982

DISPLAYING BUSINESS FORMS

Douglas Crockford

The screen is only 40 (or 38) characters wide. When you are displaying columnar data, you usually have to give up one character or more per column to separate the columns. This reduces the amount of information you can display.

But no longer! Now you can have up to eight vertical lines drawn between columns, and those lines will not take up a character position.

First, do these pokes:

```
POKE 53261,128
POKE 53262,128
POKE 53263,128
POKE 53264,128
POKE 53265,170
POKE 623,1
```

Then, determine where you want the lines and do statements like

```
POKE 53248+L,X*4+55
```

where L is a number between 0 and 7 and X is the position number where you want the line to appear. The lines will stay in place until you do a

```
POKE 53248+L,0.
```

This is using player/missile graphics to draw the lines. It requires no extra memory and does not consume any extra DMA time.

```
100 GRAPHICS 0
110 PRINT:PRINT"TYPE AWAY!":PRINT
120 POKE 764255:REM-CLEAR AWAY OLD
CODES THAT COULD BE EMBARASSING
130 PRINT "KBRD","KEY","ATSC"
140 PRINT "CODE""SMBL""CODE"
150 X=PEEK(764):REM-GET KEY CODE
160 IF X=255 THEN 150:REM-WAIT FOR
FIRST KEY TO BE PRESSED
   SO HEADINGS CAN BE READ          170
Y=PEEK(65278+X):REM-CONVERT TO
ATASCII
180 IF Y>96 AND Y<123 THEN Y=Y-32:REM-
CONVERT LOWER CASE TO UPPER
190 PRINT X,CHR$(Y),Y
200 GOTO 150
```

The second example illustrates (a) programming a loop that waits until a key is pressed allowing time for reading instructions (lines 200 and 210), and (b) program operation continuing without being halted by input operations.

EXAMPLE 2:

```
100 GRAPHICS 0
110 PRINT "COUNTING PROGRAM":PRINT
120 PRINT "KEYBOARD CONTROLS"
130 PRINT "1 - PRINT EACH NUMBER
(DEFAULT)"
140 PRINT "2 - NO PRINTING"
150 PRINT "3 - PRINT 10'S ONLY"
160 PRINT "4 - RESET TO ZERO"
170 POKE 764,255:REM-CLEAR OLD KEY
CODES
180 PRINT "PRESS ANY KEY TO START"
190 Z=1:REM-SET CONTROL TO DEFAULT
200 Y=PEEK(764):REM-CHECK IF KEY
PRESSED
210 IF Y=255 THEN 200:REM-IF CODE 255,
NOT READY
220 POKE 764,255:REM-KEY PRESSED,
CLEAR CODE
230 X=0:REM-THIS IS COUNTING VARIABLE
240 Y=PEEK(764):REM-CHECK FOR NEW
COMMAND
250 IF Y<>31 AND Y<>30 AND Y<>26 AND Y<>
24 THEN 340:REM-CHECK FOR KEYS 1-4
260 SOUND 0,50,10,10:REM-START CHIRP
270 IF Y=31 THEN Z=1:REM-DECODING
280 IF Y=30 THEN Z=2:REM- COMMAND
290 IF Y=26 THEN Z=3:REM- CODE
300 IF Y=24 THEN Z=4:REM-KEY BY KEY
310 PRINT "CONTROL ";Z:REM PRINT TO
SCREEN RESPONSE
320 SOUND 0,0,0,0:REM-END CHIRP
RESPONSE
330 POKE 764,255:CLEAR KEY CODE
340 X=X+1
350 IF Z=2 THEN 240
360 IF Z=4 THEN Z=1:X=0
370 IF Z=1 THEN PRINT X:GOTO 240
380 Y=X/10
390 IF Y=INT(Y) THEN PRINT X
400 GOTO 240
```

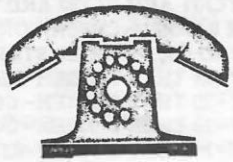
SHORT BUT USEFUL

```
10 REM LIST.ENT
20 REM BY PAUL WHEELER
30 REM
40 REM SAVE THIS PROGRAM AS D:LIST.ENT
50 REM TO LOAD,LIST,DELETE,ENTER AND
60 REM SAVE A PROGRAM TO CONSERVE ON
70 REM MEMORY, AND HELP PREVENT SLIP-
80 REM UPS IN THE PROCESS.
90 REM USEFUL FOR ELIMINATING NO
100 REM LONGER NEEDED VARIABLES THUS
110 REM SHRINKING PROGRAM SIZE
120 REM
130 PRINT CHR$(125):DIM DN$(15):PRINT
140 PRINT :PRINT "ENTER '0' TO RETURN TO
DOS:"
150 PRINT "ENTER '1' TO END:":PRINT
160 PRINT :PRINT "DISK FILE NAME:
":INPUT DN$:PRINT CHR$(125)
170 IF DN$="1" THEN END
180 IF DN$="0" THEN DOS
190 PRINT
200 PRINT "LOAD ";CHR$(34);DN$;CHR$(34)
210 PRINT :PRINT :PRINT "LIST
";CHR$(34);DN$;CHR$(34)
220 PRINT :PRINT :PRINT :PRINT "NEW"
230 PRINT :PRINT :PRINT "ENTER
";CHR$(34);DN$;CHR$(34)
240 PRINT :PRINT :PRINT "SAVE
";CHR$(34);DN$;CHR$(34)
250 PRINT :PRINT :PRINT "RUN
";CHR$(34);"D:LIST.ENT";CHR$(34)
260 REM
270 REM POSITION CURSOR TOP OF SCREEN
280 REM
290 POKE 84,0
300 END
310 PRINT :PRINT "PRESS [RETURN] AFTER
COMPLETION"
320 PRINT "OF EACH COMMAND."
```

WHEN YOU'RE LOOKING FOR

LOW PRICES, IN-STOCK MERCHANDISE, EXPERTISE

(C) = CASSETTE
(K) = ROM CARTRIDGE
All other software is on disks



CALL

JOHN

or

JAN

(202) 364-0273

OVER 1000 PRODUCTS AVAILABLE FOR YOUR ATARI®

We deliver to your ATARI® club meeting; Call for Details.

- VISA, MC add 3% (include card no. and expiration date).
- Allow 3 weeks for personal check to clear.
- COD's require 10% deposit—all COD charges to be paid by customer.
- Prices subject to change without notice—all items subject to availability.
- Money order, certified check, cashier's check or bank wire deposit accepted.
- Include full telephone number on all mail orders.
- Include $2.50 for postage and handling per software shipment (UPS).
- DC residents add 6% tax (if tax exempt, include number).

MasterCard VISA

Description	List $	Sale $
GAMES		
Adventure International		
Adventure 1, 2, & 3	39.95	29.95
Adventure 4, 5, & 6	39.95	29.95
Adventure 7, 8, & 9	39.95	29.95
Adventure 10, 11, & 12	39.95	29.95
Starfight	24.95	18.95
Rear Guard	24.95	18.95
Preppie	29.95	22.95
War	24.95	18.95
Arcade Plus		
Arcade Plus Football	34.95	27.95
Ghost Hunter	34.95	27.95
Arcade Baseball	34.95	27.95
Night Rally	34.95	27.95
Automated Simulations		
Temple of Apshai	39.95	29.95
Star Warrior	39.95	29.95
Crush, Crumble, & Chomp	29.95	22.95
Dragon's Eye	29.95	22.95
Curse of Ra	19.95	15.95
Avalon Hill		
Empire of the Overmind	35.00	27.95
Tanktics	29.00	21.95
Legionaire (C)	35.00	27.95
Broderbund		
David's Midnight Magic	34.95	27.95
Deadly Secrets	34.95	27.95
Star Blazer	31.95	24.95
Track Attack	29.95	22.95
Budgeco		
Raster Blaster	29.95	22.95
Cavalier		
Bug Attack	29.95	22.95
DataSoft		
Tumble Bugs	29.95	22.95
Pacific Coast Highway	29.95	22.95
Shooting Arcade	29.95	22.95
Gebelli		
Andromeda	34.95	27.95
Doctor Goodcode's Cavern	29.95	22.95
Infocom		
Deadline	49.95	39.95
Zork I	39.95	29.95
Zork II	39.95	29.95
IDSI		
Pool 1.5	34.95	27.95
On-line Systems		
Crossfire	29.95	22.95
Jawbreaker	34.95	27.95
Missing Asteroid	24.95	18.95
Wizard & Princess	32.95	25.95
Mousekattack	34.95	27.95
Ultima I	39.95	29.95
Roklan Corp		
Gorf	39.95	29.95
Wizard of Wor	39.95	29.95
Gorf (K)	49.95	39.95
Wizard of Wor (K)	49.95	39.95
Sirius		
Cyclod	29.95	22.95
Snake Byte	29.95	22.95
SSI		
Shattered Alliance	39.95	29.95
Battle of Shiloh	39.95	29.95
Tigers in the Snow	39.95	29.95
Synergistic Software		
Warlock's Revenge	35.00	27.95
Probe I	34.95	27.95
Voyager		
Battle Trek	39.95	29.95

Description	List $	Sale $
ATARI® COMPUTERS		
800 16K	899.00	659.95
810 Disk Drive	599.95	479.95
830Modem	199.95	149.95
850 Interface	219.95	169.95
Paddles (pr.)	21.95	17.95
Joystick (pr.)	19.95	16.95
400 16K	399.00	289.95
410 Recorder	99.95	79.95
825 Printer (80 Col.)	799.95	599.95
PLUS		
All ATARI® Accessories & Software at Exceptional Discounts		
MEMORY EXPANSION		
ATARI®		
8K RAM	49.95	39.95
16K RAM	99.95	83.95
Axion		
32K RAM	149.00	109.95
128K RAM DISK	569.00	429.95
ACCESSORIES		
DataSoft		
Le Stick	39.95	31.95
Color Print Refill	15.95	12.95
Versa Computing		
Versa Writer Graphics Tablet	299.00	249.95
BUSINESS SOFTWARE		
ATARI®		
Dow Jones Investment Evaluator	129.95	94.95
Personal Finance Management	74.95	59.95
Word Processor	149.95	109.95
Datasoft		
Text Wizard	99.95	79.95
LJK		
Letter Perfect	149.95	109.95
Mail Merger	29.95	22.95
Data Perfect	99.95	83.95
Letter Perfect w/Mail Merger (K)	249.95	189.95
Synapse Software		
Filemanager 800	99.95	79.95
VisiCorp		
VisiCalc	250.00	179.95

Description	List $	Sale $
UTILITIES		
ATARI®		
MicroSoft Basic	89.95	69.95
Macro Assembler & Text Editor	89.95	69.95
PILOT (K)	79.95	63.95
DataSoft		
Disk Detective	29.95	22.95
Datasm 65 2.0	89.97	69.95
Micro Painter	34.95	27.95
Color Print	39.95	29.95
Lisp Interpreter	149.95	109.95
Graphics Master	39.95	29.95
Graphics Generator	24.95	18.95
Basic Compiler	99.95	79.95
LJK		
Edit 6502 (K)	199.95	149.95
Optimized Systems Software		
AS/A+	80.00	59.95
BASIC A+	80.00	59.95
OS/A+ & BASIC A+	150.00	109.95
Speed Read Plus	59.95	49.95
Quality Software		
QS Forth	79.95	59.95
Synapse Software		
Disk Manager	29.95	22.95
United Software of America		
ATARIoWorld	59.95	49.95
3-D Supergraphics	39.95	29.95
Versa Computing		
Graphics Composer	39.95	29.95
ATARI® VCS		
Activision		
All $22.95 games are $17.95		
All $31.95 games are $23.95		
Imagic		
All $24.95 games are $18.95		
All $31.95 games are $23.95		
Spectravision		
All $26.95 games are $20.95		
All $27.95 games are $21.95		
BLANK 5½" DISKS		
Elephant		
SSSD		21.95
SSDD		25.95
Maxell		
SSSD		36.95
SSDD		36.95
Dysan		
SSSD		37.95
SSDD		39.95
Verbatim		
SSDD		25.50

COLUMBIA COMPUTER SYSTEMS

LOW PRICES, IN-STOCK MERCHANDISE, EXPERTISE

The largest & oldest supplier of Computers, Peripherals, & Software
in the Washington, D.C. Area

Mail your orders to: P.O. BOX 40813, Washington, DC 20016
or call (202) 364-0273 any time, day or night

CURRENT NOTES
2159 Golf Course Drive
Reston, Virginia 22091