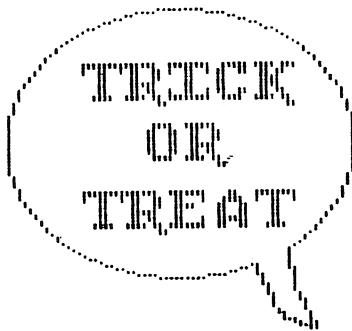


# A.C.E.C. 卍<sup>®</sup>

## Newsletter of the Atari Computer Enthusiasts of Columbus

VOLUME 4, NO 10

October, 1986



**PUMPKIN  
FROM PRINT SHOP**

### **INSIDE THIS ISSUE:**

**The Editor's Column  
Node Select: The Source Code  
September SIG Notes  
Build a Speech Synthesizer  
World Karate Championship Review  
Assembly Language Tutorial  
Foreign Facts  
Publication Library Update  
BASIC Printing Techniques  
Antic On-Line and More!**

This newsletter is written and published monthly by the Atari Computer Enthusiasts of Columbus, Ohio (ACEC). ACEC is an independent, non-profit organization interested in exchanging information about any and all Atari Home Computer Systems.

Meetings are held on the second Monday of each month at 7:15 p.m., at DeSales High School (on Karl Road, just south of Morse Rd.), and are open to the public.

Dues are \$12.00 per year, and entitle members to all club benefits (Newsletter, Disk of the Month, Publications Library, SIG meetings, group discounts at selected area merchants, etc.).

The ACEC Newsletter welcomes contributions of articles, reviews, editorials and any other material relating to the Atari computers, or compatible hardware devices and software packages.

PRESIDENT:

Sheldon Wesson  
444 N. Pearl Street  
Granville, OH 43023  
1-614-587-2716

DISK LIBRARIAN:

Charles Lusco  
Channing Terrace, #C  
Columbus, OH 43232  
614-863-4016

PUBLICATIONS LIBRARIAN:

Bill Morgens  
4943 Botsford Drive  
Columbus, OH 43232  
614-861-2431

MEMBERSHIP CHAIRMAN:

Tim Adcock  
7544 Satterfield Road  
Worthington, OH 43085  
614-764-9492

ACEC is not affiliated with the Atari Corporation; "ATARI" and the "Fuji" symbol are registered trademarks of the Atari Corporation. All other trademarks, copyrights and service marks are registered with their respective owners.

The statements expressed in this newsletter are solely the opinions of the authors, and do not necessarily reflect those of ACEC, its officers or its members. Material contained in this newsletter may be reprinted provided credit is given to both ACEC and the author(s).

The cover of this month's newsletter was printed with a Star SG-10 dot matrix printer, using XLent Software's TypeSetter 130. The newsletter itself was also printed with a Star SG-10 printer in elite pitch, using Batteries Included's PaperClip version 1.2X on a 256K RAM modified Atari 800 XL.

Address all mail to:

A.C.E. of Columbus  
P. O. Box 849  
Worthington, OH 43085

VICE-PRESIDENT:

Warren Lieuallen  
1652 Hess Boulevard  
Columbus, OH 43212  
614-488-3977

NEWSLETTER EDITOR:

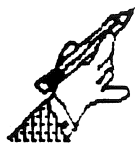
Mark Shuter  
510 Pin Cherry Lane  
Worthington, OH 43085  
614-885-0663

TREASURER:

Bill Eckert  
6632 Lisamarie Road  
Columbus, OH 43229  
614-891-9785

SECRETARY:

Norman Knapp  
1222 Norton Ave.  
Columbus, OH 43212  
614-291-2849



## The Editor's Column

Once again, I am filling in as Guest Newsletter Editor. No one called to say that they wanted time for equal billing on the campaign trail, and I figured as long as I was all set and geared up to do it anyway . . . .

I mentioned last month that I still had several ideas in the works for our newsletter. Well, here are most of them:

1. First of all, I think that with more than 120 members, we ought to be able to think up a better name for the ACEC newsletter than The ACEC Newsletter. Notice, however, that I said "we". I am hereby announcing the Name the Newsletter Contest. The rules are simple: think up a catchy and appropriate name for our newsletter, and let me know. Give me a call, mail me a letter, post a message on the club's BBS (268-0405), hire a sky-writer. Any and all entries will be considered, and the final decision will be made by the ACEC officers. The author of the winning entry will win an exciting and valuable prize (to be determined at a later date!). Unless the entry requests otherwise, I will also print all entries in the first newly-named newsletter, so if the club's membership as a whole strongly objects to the new name, you can all see the other possibilities. So, please, let your creative juices flow. If I don't get any entries, you're all going to have to live with the name that I think up, and mine will probably be pretty corny. You have been warned.

2. Second, I would like to again emphasize that all ACEC member submissions to this newsletter (whatever it's called) are always welcome, and usually begged for. We are extremely fortunate this month to have at least four separate articles from three

different members! That's probably the new ACEC record! Any material relating in any way to computers and their usage (preferably, but not necessarily, Atari computers) is desired. Your article doesn't have to be a twelve page thesis on intricacies of programming that would make Chris Crawford do cartwheels. Any topic, any length, at any level would be welcome. Material at a very introductory level is always in demand, as this newsletter is often evaluated by our prospective new members. In my order of preference, you may submit the material on a disk (which will be returned or reimbursed for), over the phone if you've got a modem, or you can even give me printed hard copy in an emergency. If possible, please use either PaperClip, AtariWriter or AtariWriter Plus (however, I personally have nearly a dozen word processors, and as they say, beggars can't be choosers!). If you are concerned about which month your material will appear, my deadline for the each month's newsletter is the first Saturday of the month of issue.

Here is a list of some of the topics that I think would make excellent newsletter articles (why don't you pick one and write something?!):

- The G: Device
- The Print Shop Companion
- The Print Shop Interface
- The F:R: Connection
- The Express! Family of Terminal programs
- The new version of PaperClip with SpellPack
- BASIC Compilers (if there are any left in production)
- Any of the Public Screen Dump programs
- The XM301 Modem
- The XMM801 Printer
- The Entire XLEnt Software series (viz. TypeSetter, Page Designer, Rubber Stamp)
- The new ICD MIO device
- Any of the many music systems available (AMS, Pokey Player, etc.)
- Rambrandt and Atari Artist
- Digitized Photos and Sound

The list goes on, but I think this is a good start. I'm ready whenever you are!

3. Related to this plea, I would also like to suggest the formation of several definite columns in the newsletter, and therefore, several columnists to write them. I think that every month, we could stand to have an article on the following topics:

- BASIC programming / Tips
- New Software / Games
- ST material
- SIG meeting Report
- ACEC meeting Report (minutes) ?
- Treasurer's Report ?
- Officer's Meeting Report ?
- BBS updates
- Librarian's Corner

You get the idea. There are several other columns which I intend to write myself, such as the Editor's Column (this drivell!), Antic On-Line, material from other users' group's newsletters (which I wouldn't have to do nearly as often if this club generated more articles [hint, hint!]), and either ST SIG update or regular SIG update, depending on which I attend.

4. I would also like to begin a Letters to the Editor column, but that obviously requires some letters to the editor. If you have some particular gripe, or compliment, or just a question, write me a note. I'll either answer it myself, or more likely, forward it to someone who can answer it. Other possibilities hinge on the fact that I am the official liason between this club and Batteries Included, and am quite active with several officers at OSS. I also serve on the Columbus CompuServe Advisory Board. So, it's up to you now. If I get some letters, I'll publish them, along with their answers.

5. Finally (for this month anyway), I would like to remind all ACEC members that "advertisements" for both hardware and original software are always welcome. This is in conjunction with our monthly

"Swap Table". If you've got some used item that you'd like to sell, let me know. I'll post it up here, and the price is hard to beat! Advertisements from local retailers is also a possibility, although I'll need some member and officer feedback before this becomes a reality.

Warren Lieuallen

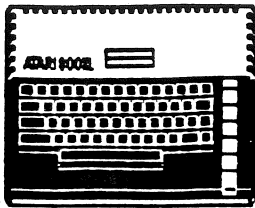


**P.S.**

As you will soon see, there is quite a bit of blank space throughout this newsletter, due to the double column printing, and the difficulty I have in getting everything to fit evenly on a page (all right, I can see where 80 column screens do, on occasion, have certain advantages! But, can they do double columns on the screen? That's what I really need.).

I have tried to fill in as much of this space as possible with my cutesy little Print Shop cards, but how much can one guy do? And I imagine you're going to get pretty tired of seeing my idea of a space-taker after a few months.

So, why not write a short article for me, and put us both out of our misery?! Even a simple little Atari Aid tip would be VERY appreciated.



# Eight-Bit Atari Programming

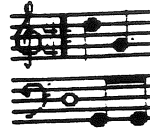
## MODE SELECT: The Source

This article is going to be tied in with my program on this month's Disk of the Month, which is called MODESEL.BAS. This program will allow the user to choose different graphic modes, and will then fill the screen in that mode with characters chosen by the user.

This program uses a machine language subroutine. As a part of my series on machine language, I will be talking about the source program for this machine language routine. This source code was created by using the Atari Assembler Editor cartridge.

```
10      *=$680      ;store code at page 6
20 SCRNL0 = $CB      ;page 0 pointer
30 ;GET VALUES FROM BASIC
40      PLA        ;remove useless value
50      PLA        ;same as above
60      PLA        ;same as above
70      STA MODE    ;store value into
variable mode
80      PLA        ;see line 40
90      PLA        ;same as above
0100    STA CHAR    ;store value into
variable char
0110    JSR OPEN    ;set up screen that
user requested
0120 L00P LDA CHAR  ;character is from
the basic program
0130    PHA        ;push it onto the
stack register
0140    ;calculate beginning of screen
memory
0150    LDY #0     ;set y register
to 0
0160    LDA $58    ;low byte pointer
to address of upper left of screen
0170    STA SCRNL0 ;store into page
0 pointer
0180    LDA $59    ;high byte
pointer to address of upper left of
```

```
screen
0190    STA SCRNL0+1;store into high
byte of page 0 pointer
0200    PLA        ;pull the value
off the stack
0210 ;fill the screen with char that user
chose (taken from the basic part of the
program
0220 STORE STA (SCRNL0),Y;store it into
address pointed to by scrnl0 plus y
0230    INY        ;increment the y
register
0240    BNE STORE  ;branch to label
store if y is not 255
0250    INC SCRNL0+1;increment the
high byte on overflow
0260    LDX SCRNL0+1;store high byte
into x register
0270    CPX #$A0   ;check to see if
high byte=end of screen memory
0280    BNE STORE  ;go back to label
store if no match
0290 ;wait until user hits proper key
0300 CKRKEY LDA 764 ;internal key
board code
0310    CMP #12    ;return key?
0315    BEQ CLOSCRN ;close the screen
if equal
0320    CMP #6     ;left arrow?
0325    BEQ CLOSCRN ;same as 315
0330    CMP #7     ;right arrow?
0335    BEQ CLOSCRN ;same as 315
0340    CMP #14    ;up arrow?
0345    BEQ CLOSCRN ;same as 315
0350    CMP #15    ;down arrow?
0360    BNE CKRKEY ;go to label
ckrkey if no match(keep looping)
0370 ;close screen
0380 CLOSCRNLDX #$10 ;set x register
for proper channel offset (channel 1)
0390    LDA #$C    ;close channel
command
0400    STA $342,X ;store command
here
0410    JSR $E456  ;perform iocb
routine
0420 ;clear key press and return to basic
0430    LDA #$FF   ;255
0440    STA 764    ;store into key
board code to clear values
0450    RTS        ;return to basic
0460 OPEN LDX #$10 ;set x register
to offset for channel 1
```



## SIG Notes

```
0470      LDA #3      ;open channel
command
0480      STA $342,X  ;store command
here
0490      LDA #SCREEN&255 ;low byte of
address of variable screen
0500      STA $344,X  ;point to address
for data to process
0510      LDA #SCREEN/256 ;same as 490
but high byte
0520      STA $345,X  ;same as 500 but
high byte
0530      LDA #C      ;full screen read
and write command
0540      STA $34A,X  ;aux.command goes
here
0550      LDA MODE    ;this specifies
which graphic mode
0560      STA $34B,X  ;store into
aux.byte#2
0570      JSR $E456   ;go to routine
that performs iocb operation
0580      RTS        ;return
0590      SCREEN .BYTE "S:" ;define variable
screen
0600      MODE .BYTE 0 ;set aside memory
for this variable
0610      CHAR .BYTE 0 ;same as above
```

When you use the `USR` command in BASIC, you may include values in the statement and push them onto the stack register. Then, when you are actually using the machine language code, you have to pull the values off the stack register to use them. You use the `PLA` command to do this.

I hope that you get a better understanding of machine language from this discussion. I also hope that you will enjoy the BASIC program that this goes with, and that you get a better understanding of the Atari graphics' modes from it.

by CHARLES W. BROWN

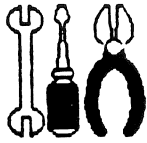
The September meeting of the ACEC Special Interest Group's (SIG's) was held on the 25th of September, at the Grandview Heights Public Library on West First Avenue. As I was not there, I am in a pretty poor position to comment on anything else!

Next month's SIG meeting will again be at the Grandview Heights Public Library, downstairs in the Meeting Room at 7:15 p.m. As always, a variety of topics will be discussed, and you're always welcome to change the subject at will. Bring your questions, bring your friends!

Luckily for us, Charles Brown kindly volunteered to write a short synopsis of the September SIG meeting, so here it is!

We had a very small turn out; in fact there was only 6 of us. This might have been due to the change in location. I went on with my attempt to give some pointers on BASIC. We had a lively talk about opening and closing files, which included the use of these commands for storing and loading data. It was explained that this method can save typing in data each time a program is run (such as a check book balancing program, etc), and can also be used for bringing a machine language file into a BASIC program. After that I tried to find a bug which had mysteriously crept into one of my programs! Dave Feeney and I tried to find it, but never did. This just goes to prove that I am no expert on the subject (*Many SIG members would argue this point, Charles! - Ed.*). As always, in my BASIC lectures, I will try to share what I know with you. This is what the SIG meetings are all about: to share information, get questions answered and solve problems. I hope that more of you will attend these informative meetings. I think that you will find them useful.

by Warren Lieuallen and Charles Brown



# Atari Hardware

## The Talk-Box by Gene Strojny

Can we talk? I mean... can we talk? Better yet, can your Atari XL home computer system talk? What? You mean you don't have a speech synthesizer for your computer? Well, now there's no excuse not to. I'm going to show you how to build one yourself for less than \$30.00. Sounds too good to be true, doesn't it? Well, it is true, using Radio Shack's SP0256-AL2 IC speech processor chip.

The main problems with the speech synthesizers currently on the market is that they cost too much, or use an excessive amount of RAM. The schematics for the "Do It Yourself" kits are usually too technically complex for all except an experienced Electronics Technician. Even the schematics for the SP0256-AL2 are obscure for someone who only knows the very basics of electronics. I have tried to simplify things and have eliminated all but the most essential components. I ended up with a schematic that just about anyone should understand.

### The SP0256-AL2 Chip

The Radio Shack SP0256-AL2 chip is a pre-programmed IC chip. It has all the basic speech sounds (*called phonemes - Ed.*) programmed into it. This means that no RAM is tied up holding the speech synthesis routines. I won't elaborate too much on how the words are formed from the basic sounds, since the SP0256-AL2 comes with a small user's manual. This manual contains a lesson on basic speech production, a list of the basic speech sounds programmed into the chip and their decimal FOKE equivalents, a small dictionary of words and their decimal

equivalents, as well as the technical data for interfacing.

## The Circuit

There is nothing critical about the circuit. You may assemble it on a perf board using wire wrapping, or if you like, you may make your own printed circuit board. The parts list is given in Table One. The wiring diagram is given in Figure One, and the pin out configuration for the joy stick ports is shown in Figure Two.

When building the circuit, install the 28-pin DIP socket first. The use of the socket is strongly recommended so that you don't have to apply the soldering iron directly to the chip and risk burning it out. Next, install the resistors, capacitors and cables. The wires in the joystick are color-coded, but as far as I know, there is no correlation between the color of the wire and the pin numbers. Check each wire to be sure which pin it corresponds to.

Table One: Parts List

<u>ID</u>	<u>Qty. and Description</u>	<u>Part #</u>
Capacitors		
C1-C4	1 4.47pF	272-121
C5	1 0.1uF 50v	272-1069
C6	1 1.0uF 16v	272-1434
Resistors		
R1	1 100K 1/4 watt	271-1347
R2	1 10K 1/4 watt	271-1335
Miscellaneous		
IC1	1 SP0256-AL2	276-1784
Xtal	1 3.579 MHz	272-1310
	1 28 pin socket	276-1997
	2 Joystick Cables	276-1978
	1 Phone Jack	274-251
	1 Plastic Case	270-222
	1 Amplifier	277-1008

Attach the wires corresponding to the joystick port's pins to the corresponding pads on the circuit board. These are labelled in the schematic diagram. Install the SF0256-AL2 chip last, after everything else is in place. Leave the chip in its package until you're ready to install it. While this kind of chip is very forgiving when it comes to miswiring, it will give out quite fast when faced with static electricity. This means that you must make certain that you are grounded and aren't charged with static electricity whenever you touch the chip.

There are two ways to hook up the audio. You may use a small battery powered audio amplifier like the #277-1008 that Radio Shack sells, or you may take your RS-232 cable end apart and solder a lead corresponding to pin #11 of the CIO port, as shown in Figure Three, to the center pin of a miniature phono jack (Pin #11 is the Cassette Audio pin.). The sound will be channeled through your monitor. If you do use this method, all other computer generated sound will be cut off until the connection is unplugged from the Talk-Box. If you plan to use the speech synthesizer in conjunction with programs that have sound effects, use the first method.

Whichever method you choose, don't (I repeat, DON'T) connect your computer to an external, unregulated line powered amplifier. If you do, you will most assuredly fry your computer into Silicon Heaven.

#### Using the Talk-Box

All that has to be done to make the chip pronounce a word is to POKE the chip's decimal addresses (via the joystick ports) with the numbers which correspond with the group of sounds which make up that word. For example, the word HELLO consists of four sounds: H/E/LL/O. After you determine the word's individual sounds, look up the appropriate decimal values in the data manual supplied with

the chip. The word HELLO would therefore be represented by the numbers 27, 7, 45, and 53.

The joystick port must be configured for output before you can POKE data to it. This is done by PEEKing location 54018, subtracting 4 from the value found there, and POKEing the result back into 54018. Then you must POKE a 127 into 54016 and then returning 54018 to its original value.

Your program must check to see if the SF0256-AL2 chip is busy before sending any data to it through the ports. This is done by checking bit 8 to see whether it's high or low (viz. 1 or 0). the chip sets this bit low when it's not busy, and high when it is busy. When bit 8 goes low, you must first POKE 54016 with (64 + the decimal value of the sound you wish produced). Next, strobe bit 7 by POKEing 54016 again with only the decimal value of your sound. This lets the chip know there is data on line, ready for it to accept. The chip will then accept the data and simultaneously set bit 8 high again until it is done making the sound.

Listing One is a small program to test your Talk-Box. Just type in the listing and RUN it after you have plugged the Talk-Box into the joystick ports. Make sure that if you are using the battery powered amplifier, it is turned on and the volume is up. When the program is RUN, you should hear it say, "Hello. How are you?" If it doesn't, check to make sure you plugged all the cables into the correct ports -- the Talk-Box won't work if they are reversed. If it still doesn't work, re-check your wiring.

If at this time you're all confused, don't worry about it. You really don't need to understand how the process works to use your Talk-Box. Just copy lines 100, 130 and 140 into your own program.

Listing Two is a small word development program. It allows you to quickly enter words so that you can



preview their sound. This is especially handy when you're not sure which of several similar sounds will produce the desired results.

When you RUN Listing Two, it will prompt you to enter the decimal equivalent of the first sound in your word. Type in the number and then press RETURN. Continue until all the sounds of your word have been entered. Then press RETURN again. The program will pronounce your word, and will ask if you would like it repeated. If not, just press RETURN to continue.

There are many uses for a speech synthesizer. Use your imagination. You might use it to give verbal error messages, prompts, or any other message that would normally be printed to the screen. I hope you enjoy the project as much as I did. If the response is good, I'll see if I can come up with some more easy projects. Did I hear someone mention a Do It Yourself printer buffer? Well, I just happen to....

FIGURE #1

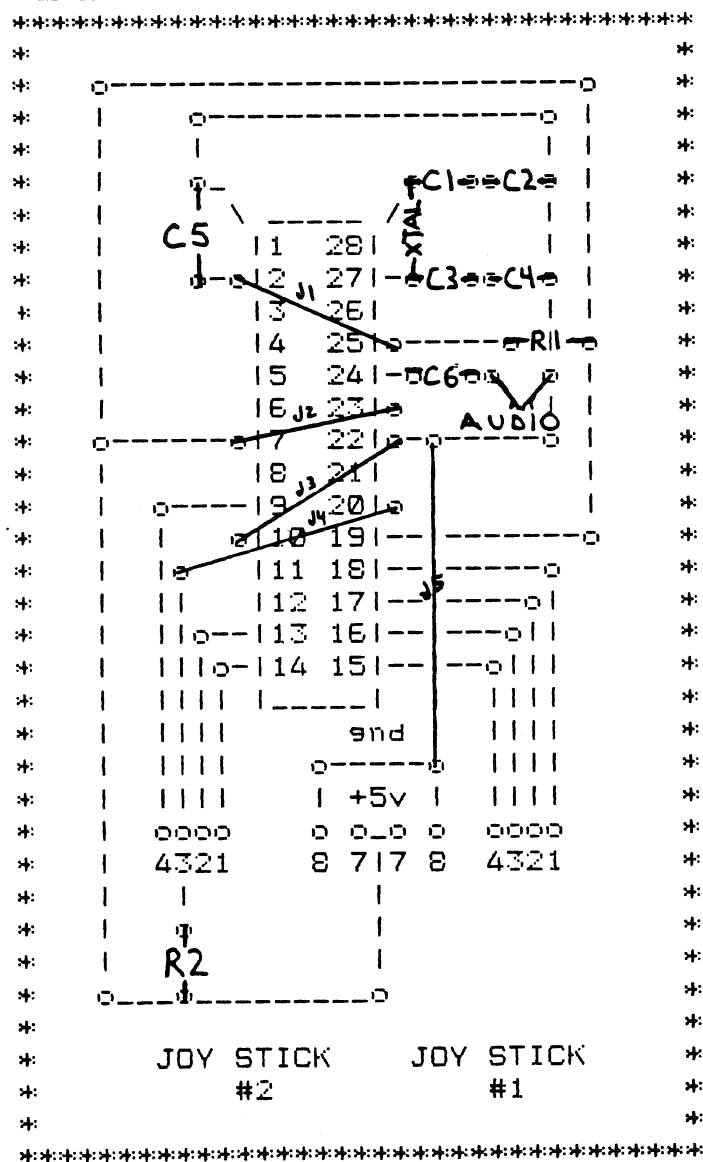


FIGURE #2

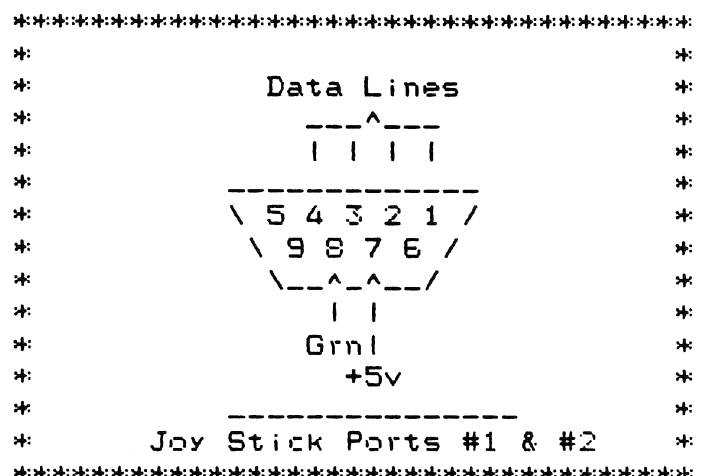
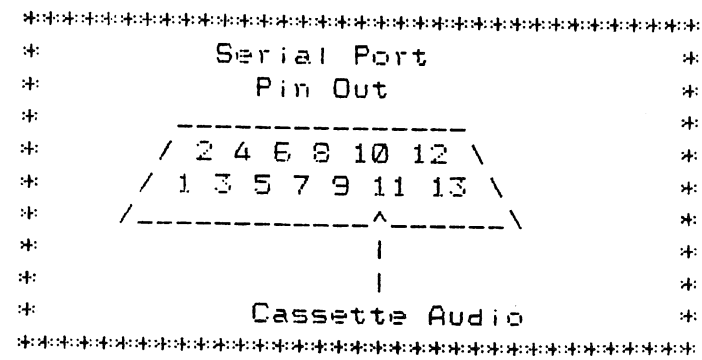


FIGURE #3

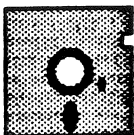


-----  
**LISTING #1**  
-----

```
100 P=PEEK(54018):POKE 54018,P-4:POKE 54016,127:POKE 54018,P
110 TRAP 500
120 FOR S=1 TO 70:READ D
130 IF PEEK(54016)>128 THEN 130
140 POKE 54016,D+64:POKE 54016,D
150 Next S
280 DATA 27,7,45,15,53,4,27,32,2,59,2,49,31,4
500 END
```

-----  
**LISTING #2**  
-----

```
10 REM THIS PROGRAM IS TO CREATE WORDS
20 REM FOR TALK BOX.
25 REM THESE LINES SET UP THE PORTS
30 DIM A(20),T$(1)
40 P=PEEK(54018):POKE 54018,P-4:POKE 54016,127:POKE 54018,P
50 GOSUB 200:GOSUB 300
60 REM INPUT DEC. DATA TO TRY WORDS
70 TRAP 80:FOR X=1 TO 20:INPUT Y:A(X)=Y:NEXT X
80 POP
85 FOR S=1 TO 20
90 IF PEEK(54016)>128 THEN 90
100 POKE 54016,A(S)+64:POKE 54016,A(S):NEXT S
110 GOSUB 220
120 INPUT T$:IF T$="Y" OR T$="y" THEN 85
130 GOTO 50
199 REM PROMPT MESSAGE ROUTINES
200 ? " ENTER UP TO 20 NUMBERS ":B=17:RESTORE 400:GOSUB
260:RETURN
220 ? "REPEAT? (Y/N)":B=12:RESTORE 410:FOR DLAY=1 TO 300:NEXT
DLAY:GOSUB 260
245 RETURN
255 REM READ DATA FOR PROMPTS
260 FOR S=1 TO B:READ AA
270 IF PEEK(54016)>128 THEN 270
280 POKE 54016,AA+64:POKE 54016,AA:NEXT S
290 RETURN
300 FOR X=1 TO 20:A(X)=0:NEXT X:RETURN
399 REM DATA FOR PROMPTS
400 DATA 9,45,19,43,2,7,11,17,52,1,11,15,16,28,52,55,2
410 DATA 37,26,45,1,6,1,14,19,9,19,17,2
```



## Software Review

WORLD KARATE CHAMPIONSHIP  
BY EPYX  
review by Charles Brown

I've gotten a new game for the 8 bit Atari, and I think that it shows what these little machines can really do. I am sure that it could be a lot better on the ST's, but I am satisfied as what it can do on mine.

World Karate Championship is the best karate game that I have seen. It has excellent graphics, and also has very good sound effects -- some of the sounds make you almost feel the blows. It also features some pretty good backgrounds.

When you boot up the program, you will be sent into the Demo Mode. Here you will see the two fighters in a bout. You can then choose to play against the computer or against another player.

You have a total of 13 different moves, all of which are selected with the joystick. Some of the moves include: back spin kick, center kick, spinning lunge punch, and a low sweep. Choosing your move is done by moving the stick in one of the eight different directions, either with the button pushed, or not. As this would allow for sixteen different positions, there are several moves that can be selected by two different positions of the joystick.

One thing that makes this game different from others is the scoring system. Standing in the background of the bouts is the referee. He is supposed to control the scoring for the match. Besides getting regular numerical points like many other "arcade" games (100-1000 points per move), you are also awarded match points, just like in real karate

matches. When one of the players hits his opponent, the referee will award either a half or a full point. This depends on the difficulty of the move. If you get 2 whole match points, you win. When you score a certain amount of regular points you advance to the next level or belt. The belts go from white all the way to black.

Let's say that you hit your opponent with a jump kick. This is one of two highest scoring moves. If the referee thinks you did the kick well, he will give you a full match point. This earns you 1000 regular points. If the referee didn't think you did it particularly gracefully, he will only give you a half of a match point. This will only earn you 500 regular points. This type of scoring it makes the game more realistic.

I like the game because it is so different. It also gives you the feeling you are really there. There are some draw backs to it, however.

When you are fighting, you have to keep track of which side of the opponent you are on. When you start out you are the player on the left against the computer. If you move to the other side of the computer player, you have to move the joystick in the opposite direction from that listed in the book for each of the moves. For example, when you are on the left side of your opponent you would push the button and move the stick straight to the right for a center kick. If you were on the right side of you opponent you would move the stick to the left instead. I think this makes it to confusing.

I also think they should have provided for a choice of joystick or keyboard control. You will sometimes have trouble getting the stick in the right direction, especially with the diagonal moves. They should also let you enter your initials with the keyboard for the high scores. Instead they make you use the joystick. They also don't save

the high scores to disk for a permanent record.

Even though it has a few drawbacks, I really enjoyed this game. With the different features that I have mentioned, it makes the game seem quite realistic. It also gives you a glimpse of what the Atari's are capable of.



## In-Depth Tutorial

ANTIC PUBLISHING, INC.  
copyright 1985.  
Reprinted By Permission.

CHRIS CRAWFORD'S  
ASSEMBLY LANGUAGE COURSE - Lesson 1 of 8

### WHY LEARN ASSEMBLY LANGUAGE?

Assembly language is the great barrier that divides the professional programmer from the amateur. It is the most powerful language available for a microcomputer.

There are four reasons for learning to program in assembly language. First, the speed of execution of assembly language is very high -- about ten times higher than BASIC on the average, and perhaps a thousand times faster on certain operations.

Even ACTION!, the fastest high-level language, is only about half as fast as assembly language. Second, assembly language tends to be more compact than many languages. Again, ACTION! provides a good comparison. Code produced by ACTION! is about twice as large as equivalent assembly language. (*Other compiled languages are even worse -- Ed.*)

The third reason to program in

assembly language is that assembly gives you access to features of the machine that simply are not available in high-level languages. Interrupts are the most notable examples.

Finally, the most important reason for learning to program in assembly language is that it will help you to understand the machine better. And that is a very good place to begin, for you cannot learn assembly language unless you know a little bit about computers.

### HOW COMPUTERS WORK

I am now going to describe how computers work, in very rough terms. Computers operate on a hierarchy of concepts that spans a great range, rather like the hierarchy that starts with protons and electrons, moves through atoms, molecules, cells, people to civilizations.

A civilization of atoms is composed of protons, neutrons and electrons, but to understand how it is so composed one must know a great deal about the intermediate steps. So too is a computer composed of transistors. There are four intermediate steps between the transistor and the computer.

A transistor is an electrically operated switch. We can assemble transistors into gates that will turn circuits on or off depending on the states of other circuits. There are a variety of gates reflecting the various Boolean operations: AND, OR, NOT, NAND, NOR and EOR.

Gates can be assembled into latches, decoders, and adders. A latch is the simplest memory element: it remembers one bit of information. A decoder translates a number encoded in binary form on a few wires into a selection of one of many wires. An adder will add two one-bit values, with a carry, and generate a carry of its own.

We can next broaden each of these devices into an eight-bit device by simply slinging the devices side by side. Eight one-bit latches slung side-by-side give one byte of RAM. Eight adders make an eight-bit adder.

We can thus create a RAM module by building many bytes of RAM. We access this RAM module with three buses: a data bus, an address bus, and a control bus. The data bus carries information between the central processing unit and the RAM module. The address bus is sixteen bits wide; a decoder in the RAM module takes the numeric value on the address bus and decodes it to select the single byte of RAM that is indicated by the address. The control bus establishes the direction of the data flow on the data bus and the timing of data transfer.

The central processing unit (CPU) represents the highest intellectual level of the computer. It is composed of four parts: the Arithmetic and Logic Unit (ALU), the registers, the address bus controller, and the instruction decoder. The ALU is composed of adders and gate arrays that crunch numbers. The particular device to use is selected with a decoder.

The registers are simply on-board RAM. The address bus controller is a device that puts the desired RAM address onto the address bus. The real heart of the CPU is the instruction decoder, a very complex decoder that takes the program instructions out of RAM and translates them into action. It does this by feeding the instructions (which are numbers) into decoder circuits that activate the desired gateways in the CPU.

#### PROGRAMMING A MICROPROCESSOR

Machine code is nothing more than a bunch of numbers (0's and 1's) that mean something to the CPU. It's hard to work with pure binary numbers, so we use a little code that makes it easier for us to understand the codes that the computer uses. This programmer-friendlier code is

called assembly language. It is a direct, one-to-one translation of machine code. Here is an example of the two side by side:

<u>Machine Code</u>	<u>Assembly Language</u>
A9 05	LDA #FINGERS
133 \$9C	STA COUNT

The code on the right may not look very readable, but you must agree, it's far more readable than the code on the left. And they both mean exactly the same thing.

Unfortunately, the computer cannot read the assembly code, only the machine code. Therefore, we need a translator program that will translate the easier-to-understand code on the right into the impossible-to-understand code on the left. This translator program is called an assembler.

A program that goes in the reverse direction, translating machine code to assembly, is called disassembler. It may seem like a bother to go through all the hassle of using an assembler, but it is actually much easier.

Assembly language is not only more readable than machine code, but it is also assembly-time relocatable; this means you can move it around in RAM freely before you start the assembly process. A good assembler also offers a number of extra features that make it easier to keep track of your program or modify it quickly.

#### USING AN ASSEMBLER

There are three steps involved in writing an assembly language program: editing, assembling, and debugging. Editing is the process of typing in your assembly language statements. Assembling is the invocation of the assembler. Debugging is the process of running your program and analyzing why it doesn't work. Thus, the entire process of

writing an assembly-language process can be described by a fictitious BASIC program:

```
FOR I= 1 to 1,000,000,000...!  
EDIT PROGRAM  
ASSEMBLE PROGRAM  
DEBUG PROGRAM  
NEXT I
```

### THE 6502 MICROPROCESSOR

The first item in the 6502 that I will describe is the accumulator. This is a single one-byte register in the 6502. It is the central workbench of the microprocessor; almost everything happens in the accumulator. Your first of three instructions on the 6502 is:

LDA address (Load the Accumulator with the contents of address). This instruction loads the accumulator with the contents of the memory location specified by the value of address. The address can be specified by either an outright value, such as \$0600, or a symbolic reference, such as FISH, where the value of FISH has been previously declared by, say, an ORG statement or an equate statement.

LDA #value (Load the Accumulator with value). This is much like the earlier statement; it loads the accumulator with a number, only the number loaded is specified immediately rather than stored in a memory location. Thus, the command LDA # 9 will put a 9 into the accumulator.

STA address (Store the Accumulator into address). This command will store the contents of the accumulator into the RAM location whose address is specified in the command. It is just like the first command, except that the direction of data motion is reversed. The LDA command is like a read, which the STA is like a write.

You are now equipped to move data around inside the computer. These commands will allow you to read data from

one area of memory and store it into another. LDA and STA are the two most common instructions used in any 6502 program.

Exercise: Write a program that will read the contents of address \$FE00 and store the result into address \$680. Your biggest problem here will be just getting your assembler to work. Therefore, I will give the answer away right now:

```
ROMADD ORG $FE00  
RAMADD ORG $680  
ORG $600  
LDA ROMADD  
STA RAMADD  
BRK  
END
```

That's the program. Try to get it running with your assembler.



BE SURE  
NOT TO  
MISS

THE NEXT EXCITING  
INSTALLMENT OF  
CHRIS CRAWFORD'S  
EIGHT PART SERIES ON  
ASSEMBLY LANGUAGE  
PROGRAMMING  
ON THE ATARI COMPUTERS



## Foreign Facts From Far Away

First Annual Users' Computer Awards  
by Jeff Hasse  
From the September 1986 Pokey Press

Looking back over 1985 and much of 1986, I find it is once again time to pass out the First Annual Users' Computer Awards (*Ceremonies will be held at Hendy's after our October meeting. - Ed.*). We have seen some great stuff come out in the computer industry lately, lots of changes, and more frank garbage than ever before! If this trend continues to go unrecognized, I fear for us all. Enough said, let's get on with the awards.

**Best Computer of the Year:** The Commodore 64C - I've always wanted a C64 put in a Fischer-Price case.

**Worst Computer of the Year:** The IBM PC Portable - they have yet to sell 5,000 of these in two years (at least they could give away the PCjr's).

**Best Software Release:** This would have to go to Commodore for their amazing bouncing Boink. Every other computer in the industry seems to have copied this one.

**Worst Software Release:** Atari DOS 3.0 - it was so bad, the next version was 2.5!

**VaporWare of 1985-86:** The PCjr, the Commodore +4, the Commodore 16, and the Apple IIx.

**Best Hardware Add-On:** RamWorks/4 - I've always wanted a 4 megabyte RamDisk for my Apple IIe, especially when it's only \$1195!

**Worst Hardware Add-On:** A second disk drive for the PCjr - there isn't one.

**Best Kept Secret:** That the Commodore 1541 Disk Drive is slower than their tape recorder.

**Best Marketing Strategy:** Commodore wins hands down. You can now buy a +4 for only \$99. If you've never watched TV at 2:00 a.m., you would never know this.

**Best Price/Performance Ratio:** South Asia gets the nod for producing the IBM clones for \$395 plus shipping.

**Best Technology:** Commodore gets this for their 128. Not only can it run the "latest" CP/M programs, and 80% of the C64 stuff, but also the 6 programs written for 128 mode.

**Best Technology runner-up:** The Atari 1040 ST - Someone is finally selling \$4 memory chips for less than \$1000.

**Best Product Announcement:** Atari plans to offer a 64-bit computer, if and when the technology becomes available.

**Best Product Announcement runner-up:** IBM announces the Convertible, but everybody buys the Toshiba anyway.

**Award of Excellence:** This goes to Jack Tramiel for taking a company that lost \$500 million dollars, and turning a profit.

**Award of Irony:** This has to go to Commodore for selling more computers than any other company in the world, and losing \$35 million.

**Award of Ingenuity:** To whoever invented the mouse. This guy must be incredibly rich by now.

**Best Computer Magazine:** Macazine - They published one issue, and then went bankrupt.

**Best Computer Magazine runner-up:** Amiga World - Enough said.



# Librarian's Report

A.C.E.C. PUBLICATIONS LIBRARY

## COMPLETE LISTING

by: Bill Morgens  
October 13, 1986

Many thanks to those who donated publications at the August and September meetings (OH! My achin' back!).

These publications are available to be checked out by members only at no charge. We request that library items be returned at the meeting following check-out so that others may share. This listing will be updated periodically and printed in the newsletter.

The A.C.E.C. Publications Library eagerly solicits your donations of no longer wanted magazines and books. Also wanted are photocopies of Atari-related articles of general interest appearing in other magazines such as Computer Shopper, Byte, etc.

### ANTIC Magazine:

1982 - Apr, Jun, Aug, Oct/Nov, Dec/Jan  
1983 - Feb/Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec  
1984 - Jan, Feb, Mar, Apr, Jun, Jul, Aug, Sep, Oct, Nov, Dec  
1985 - Jan, Feb, Apr, May, Jun, Jul, Aug, Sep, Oct, Dec  
1986 - Jun, Apr, May

### ANALOG Magazine:

1983 - #10 Feb/Mar, #11 Apr/May, #12 Jul/Aug, #13 Sep/Oct, #14 Nov/Dec  
1984 - #15 Jan, #16 Feb, #17 Mar, #18 Apr, #19 Jun, #20 Jul, #21 Aug, #22 Sep, #23 Oct, #24 Nov, #25 Dec  
1985 - #27 Feb, #28 Mar, #29 Apr

1986 - #35 Aug

ATARI EXPLORER - April/May 1985

DR. DOBBS' JOURNAL - June 1985

CREATIVE COMPUTING - Sep 84, Jan 85, May 85

COMPUTER SHOPPER - Apr 86

BYTE - Sep 81, Feb 84, Jun 84, Oct 84

### COMPUTE! Magazine:

1984 - Nov, Dec  
1985 - Jan, Feb, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec  
1986 - Jan, Feb

### BOOKS:

Stimulating Simulations - C. W. Engel - Twelve unique programs in BASIC

Atari BASIC - Bob Albrecht, et al.

Inside Atari BASIC - Bill Carris

Advanced Atari BASIC Tutorial - R. A. Peck - A "SAM's" book

De Re Atari - A guide to effective programming

Atari Operating System Source Listing - Atari Corp.

Mostly BASIC: Applications for your Atari - Howard Berenbon - Books I & II

Kids and The Atari - Howard H. Carlson

Programming Techniques - Vol 2 - Simulation - Ed. by B. W. Liffick, - Vol 3 - Numbers in Theory & Practice

The Best of Creative Computing - Vols. 1, 2, & 3 - Ed. by David Ahl

Crash Course in Microcomputers - Howard W. Sams



The Book of Atari Software - 1983 & 1984

Atari Special Additions - A 1982 product catalog from Atari Corp.

The Creative Atari - Ed. by David Small, et al.

Some Common BASIC Programs, Atari Edition - Lon Poole, et al.

Woz, The Prodigal Son of Silicon Valley - Doug Garr - a biography of the developer of Apple computers.

Hands-On BASIC for the Atari - Herbert Peckham

Machine Language for Beginners - Richard Mansfield

Hackerbook for your Atari Computer - H. C. Wagner

The Atari Assembler - Don & Kurt Inman

VisiCalc for Science & Engineering - Stanley Trost and Charles Pomernecki

VisiCalc Home & Office Companion - David M. Castlewitz, et al.

Controlling Financial Performance - Dennis P. Curtin, et al.

Atari Computer Program Writing Workbook - Alan North

#### Exchange Newsletters:

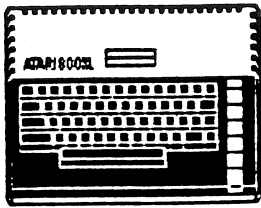
The listing of bulletins available is not a month-by-month compilation. If you are currently subscribing to an out-of-town Atari-oriented bulletin, please share your old copies with other members by donating them to the library. (As I get more address, I will be happy to send copies of our newsletter to the other groups - Ed.)

Current Notes - Washington DC area  
J.A.C.G. - Jersey Atari Computer Group  
Mile High Atari Magazine - Denver, CO  
P.A.C.E. - Pittsburgh Atari Computer Enthusiasts  
Nybbles & Bytes - Phoenix, AZ  
Milatari - Milwaukee Area Atari Users Group  
S.L.C.C. Journal - San Leandro (CA) Computer Club  
Neuron - Austin (TX) Atari Computer Enthusiasts  
F.A.C.E. World - Peninsula Atari Computer Enthusiasts (VA)  
N.O.A.U.G. News - New Orleans Atari Users Group  
The Pokey Press - Atari Computer Club of the Palm Beaches (FL)  
Blackhawk A.C.E. Newsletter - B.A.C.E. of Waterloo (IA)  
Keeping FACE - Pittsburgh Atari Computer Enthusiasts  
Flagpost - Fort Leavenworth (KS) Atari Group

We are currently exchanging bulletins with several other Atari groups around the country and there are several issues available for check-out. There are some very good articles in these exchanges and this is an excellent way to keep up with what Atarians are doing around the country. We hope to be able to publish the best excerpts from some of these bulletins. (Look for my Foreign Facts column in each month's issue -Ed.)

# WANTED!

NEWSLETTER ARTICLES,  
REVIEWS, EDITORIALS,  
ARTWORK, ETC.



# Eight-Bit Atari Programming

## TO PRINT OR TO POKE?

The graphic capabilities of the Atari are very powerful; they are unmatched by any other computer in its price range. However, there are some unusual requirements to use them. In this article I will try to explain some of them.

There are several different ways to put a character on the screen. The first and simplest way is to print it directly. To do this, you simply use the PRINT "x" command, and put your character inside the quotation marks. Although easiest to understand, this is also the slowest version.

Another way to print is by using the PRINT CHR\$(n) command, where n stands for the ATASCII code for the desired character. Let me discuss this ATASCII code briefly.

The PRINT CHR\$(n) command tells the computer to print the character which corresponds to the number inside the parentheses. This command can be used to access all of the 256 characters in the Atari computer (which are numbered from 0 to 255 - Ed.). Nonetheless, you will still be printing the letters to the screen and as a result it will be slow.

The other option to PRINTing is to POKE a value for the character that you want printed directly into the screen memory location. This is the fastest way; it is also the most confusing. Before you can start POKEing your values you have to know where to put them in memory. So, you have to know the memory locations for your desired graphics mode. You can find this out by using the following basic command once you are in

your particular graphics mode:  
? PEEK(88) + PEEK(89)\*256.

This will give you the memory location for the upper left corner of your screen. By knowing this, you can figure out where you want to put your values.

Now you need to know what value to use in order to get the character that you want. It would be really easy to use the ATASCII code for the value, but unfortunately, that won't work. There are several different sets of codes for the numbers in the Atari. One of them is the internal keycode. It is this code which is used to POKE the characters directly to the screen memory. If you try to POKE the ATASCII number to the screen instead, you will end up with the wrong characters being displayed. I have included a conversion chart for your convenience.

ATASCII	to get <u>KEYCODE</u>
0-31	+64
32-95	-32
96-127	no change
128-159	+64
160-223	-32
224-255	no change

If you use this chart you will be able to determine what number to use to get character you want. The program MODESEL on this months Disk of the Month shows how these numbers are used. If you list out lines 3000-3999 of that program, you will see the conversion table that I used. You can use it for your own applications as well. Remember, if you want to print the character use the ATASCII value. If you want to POKE it to the screen, use the internal keycode value. I hope that this will help you out a little.

by CHARLES W. BROWN

# **A** Antic On-Line

ANTIC Publishing, Inc.  
copyright 1986. Reprinted by  
permission from Antic Online.

Atari Bits And Pieces  
by Gigi Bisson

## INDUS IS OUT

Prospective Indus disk drive owners be warned; the Chatsworth, CA based peripherals manufacturer is no longer in business. A company called National Logic took over the new Indus MIDI sequencer product line, and Future Systems, Inc. purchased the rights to the Indus GT Atari and Commodore compatible floppy disk drive line.

Future Systems has notified registered Indus owners that their Indus warranties are no longer valid. A service contract can be purchased from Future Systems for \$24.95.

According to Future Systems President Gary Grewal, the company intends to release several products Indus had in the works before the reorganization, including a CPM/BIOS RAM charger that plugs into existing disk drives. The new address is: Future Systems, 9811 Owensmouth, Suite 9, Chatsworth, CA 91311. (818)407-1647.

## MIRAGE FILES CHAPTER 11

Mirage Concepts, makers of the H & D Base database and other ST products in the Holmes & Duckworth Micronomists software line, have filed for Chapter 11 protection.

Due to their inability to fulfill a \$175,000 business loan, their bank

"physically appropriated the company's assets on May 21", Mirage Vice President Michael Reinhold said in a prepared statement mailed to creditors. As their Fresno, CA offices have been shut down, company officials were not available for comment.

## ACTIVISION/INFOCOM MERGE

Activision, the Mountain View, CA entertainment software company, has reached an agreement to merge with Infocom, the Cambridge, MA interactive fiction game developer.

Activision Chairman and Chief Executive Officer James Levy said that Activision will exchange 2.0 million shares of Activision common stock, valued at \$7.5 million, for all outstanding shares of Infocom stock.

Infocom will maintain separate product development and marketing operations in Cambridge. The merger will not be official until a final agreement is signed on June 30, 1986.

Activision has also recently acquired Gamestar, creator of Championship Golf, and GBA Championship Basketball, two ST games slated for Fall release; and Creative Software, a productivity software company. Activision was founded in 1979 when five dissatisfied employees left Atari Inc. to form their own VCS game cartridge company.

## ATARI MORALITY

A man in Georgia reportedly programmed his Atari 800 and modem to call the Jerry Falwell toll-free fund collection number every 38 seconds. Apparently he was attempting to prevent the Moral Majority from getting tax-free contributions which, the computerist contended in an interview with Cable News Network, is itself immoral.

Southern Bell/AT&T traced the offending calls and ordered the

anti-Falwell Atariian to cease and desist, threatening to discontinue his telephone service. (Credit for this morsel of Atari trivia goes to the Jersey Atari Computer Group newsletter.)

### REMOTE CONTROL PETSTERS?

Nolan Bushnell, Atari Inc. founder, and current President of Axlon Inc., the company responsible for Pesters, those adorable fur-covered robots, is rumored to be considering a merger with Steve Wozniak's CL-9 company. Axlon would acquire Woz's remote control interface venture in a stock swap that would involve no cash. Teddy Ruxpin, watch out.

### MORE PEOPLE USING ATARIS

Judith Cohn is conducting Cell Image Analysis research at NASA with the aid of an Atari ST computer and the DEGAS graphics program from Batteries Included. Cohn is currently writing an ST program which she hopes will help her partition digital images of human reticulocytes (individual cells which form a network of cellular tissue in the human body) into two groups.

### FREE PUBLICITY

Trying to sell your product to the education market? Put it aboard the Teacher Resource and Computer Training Center, a roving computer "mobile unit" equipped with Atari ST and Apple II computers, textbooks, software, comfortable couches and, of course, hot coffee.

The mobile unit cruises the parking lots of 25 school districts in the Wayne-Finger Lakes area of upstate New York, giving 2700 teachers hands-on experience with educational software and computers.

To get your Atari ST educational product on the bus, send a complimentary sample of software conspicuously marked with the name and address of your sales representative, to: J.C. Crawford,

Wayne-Finger Lakes Area Teacher Resource and Computer Training Center, 3501 County Road 20, Stanley, NY 14561 (716) 526-6431.

### MIDI MAGIC

Sweet symphonies drifted through the Antic offices when Carl Bacani of Computer Support, our friendly, neighborhood Atari dealer, popped in to show off his MIDI musical compositions.

It sounded better than live music or any recording -- sharp, clear, each note distinct. Yes, you too can create stunning music with Activision's \$49 Music Studio program, and \$5,000 worth of hardware.

Bacani's MIDI set-up included an Atari 1040ST computer hooked up to a Roland TR-727 Rhythm Composer, Casio CZ-101 synthesizer, Yamaha RX 15 Digital Rhythm Programmer, Yamaha TX 7 FM expander, Casio TB-1 MIDI Thru Box, and a BOSS BX 600 6-Channel Stereo Mixer. An FM stereo receiver and two speakers amplified the sound.

Bacani will help Atari Corp. "unofficially" display the ST's MIDI prowess at The Bay Area Music Fair, sponsored by Electronic Musician Magazine, Windham Hill Productions, and several of big stereo dealerships and synthesizer companies.

Focusing on MIDI and digital music, the three-day show features live music performances, workshops and booths. Highlights include a speech by music publicist Augie Blume, and seminars by such notables as Gary Leuenberger, a synthesizer expert who has worked with Quincy Jones, Julian Lennon, Toto and Billy Joel. Hybrid Arts will be showing off the Atari ST-compatible DX Droid at the fair.

Admission is \$4.75 at the door, which is located at the Showplace Square in San Francisco; June 27, 28 and 29. For information, call (415) 864-2333. Watch

for a report on the show here on Antic Online.

#### WHY DVORAK LEFT

Don't turn to the last page and expect to find that familiar bold-faced type in InfoWorld.

John Dvorak, long-time InfoWorld institution and once editor of the publication no longer graces the pages of the weekly microcomputer industry tabloid.

Sources close to InfoWorld say Dvorak was unhappy with the publication's recent emphasis on the dull business market in an attempt to woo the MIS (Manager of Information Services) reader. MISs make major microcomputer purchases within corporations.

Dvorak continues his controversial column in Computer Currents, a California weekly tabloid, and writes a non-computer column for the San Francisco Examiner.

#### JACK'S A TOYS R US KID

The Atari invasion in the nationwide Toys R Us retail toy store chain seems to finally have taken effect. Our roving shopper reports seeing an entire shelf in a local Toys R Us store dominated by 8 and 16-bit Atari hardware and software.

#### DID YOU KNOW?

"Hackers" author Steve Levy is rumored to be working on a Broadway musical based on the M.I.T. programmers of the 60's called "Leader of the Hack." Rather than act, the actors will sit around eating Chinese food and discuss different ways to make the theater's sound system better.

Chris Crawford's latest game: "Balance of Checkbook".

The plan to stop software pirates by joining forces with National Child Watch, the folks who print those messages on

milk cartons.

Or how about buying software at 7-11? "I can't wait to get technical support from the convenience store staff...Give me four Twinkies, two of those Beef & Bran burritos, and tell me, when I'm doing an asynchronous download in XMODEM, do I have to adjust the parity bit to Odd or Even?" writes Frank Simon, in a parody editorial.

These and other humorous, high-tech pseudo-news stories can be found in Comedy By Wire: The Computer/Comedy Newsletter. A one year subscription to the monthly, four-page, publication is available for \$9 from: Comedy By Wire, 431 West 45th Street, New York, NY 10036.

\*\*\* ATTENTION !! \*\*\*  
\*\*\* Sept. 19, 1986 \*\*\*

#### ATARI GOES PUBLIC!

Atari Corp. has filed a registration statement with the Securities Exchange Commission for a public offering of stock in this formerly "closed corporation."

The initial offering will consist of 4.5 million shares at an offering price of \$11.50 to \$13.50 a share. This will represent a 16% control of Atari Corp., and should raise between \$51.8 million and \$60.8 million.

Atari needs this capital to pay off the \$36.1 million debt it still owes to Warner Communications Inc. In addition to a cash reimbursement, Atari also plans to offer Warner 7.1 million shares, or about 25% of the 28.3 million that are planned for this initial offering, leading to a 4% share of the control for Warner.

It is expected that the stock offering will be an attractive one. According to the prospectus, Atari Corp. has made an impressive turn-around. Starting from a \$26.7 million loss in 1985, the company now boasts a \$12.4

million profit for the first half of 1986. Tim Ejarin, a computer industry analyst for Creative Strategies Inc. estimates that this translates into total sales of approximately 260,000 computers. Other brokers and analysts have stated that, based on the current financial status of Atari Corp., "the price is fair, and the timing is right."

A recent story in the Wall Street Journal reflects this, and helps to spread the word to the public at large.

How will public ownership affect Atari Corp.? Let the world know what you think! If each CompuServe Atari#SIG member bought 100 shares, we could CONTROL Atari!





**ATARI COMPUTER ENTHUSIASTS  
OF COLUMBUS**

**UPCOMING MEETINGS:**

**ACEC**

**November 10**

**December 8**

**SIG's**

**October 30**

ACE of Columbus Newsletter  
Dr. Warren Lieuallen, Editor  
1652 Hess Boulevard  
Columbus, OH 43212

**FIRST CLASS MAIL**

**To:**