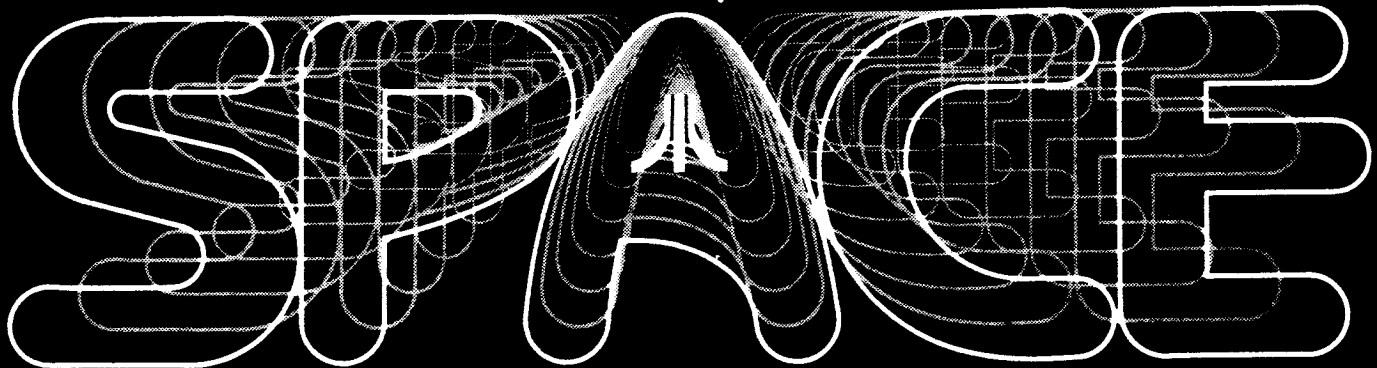


Saint Paul ATARI Computer Enthusiasts



An independent computer user group

JULY - 1985

BYTES FROM THE PRES.

by Bob Floyd

New Vendors - The Computer Supply Store, 735 N. Snelling Avenue (about 1.5 miles south of our meeting place) is offering a 10% discount on all supplies to SPACE members. You must show them your membership card. Softcost Software will now be selling software at a discount at our meetings.

ST520's - Has anyone ordered an ST yet? If ~~you~~ please volunteer to bring it to the meeting. Perhaps a vendor would demo one for us. I'm personally not quite ready to buy one yet, although I'm barely restraining myself. I'm waiting to see an ST with software, first hand, before buying one. I'm also wondering about the operating system on a ROM chip, possible 1000K disk drives, built-in disk drives, etc. Does anyone know whether the Atari color monitor supports 640x400 resolution (monochrome) when the ST is in this mode? With these questions bothering me, I'm not sure how to convince the membership to purchase an ST for the club just yet. Do you feel differently? Be sure to let me know during the meeting if you do.

Beginner's Group - This mini-group will again meet in room B-36 after the meeting. The 400 computer will be used.

Thank-you's - Thanks to Steve Pauley for volunteering to be temporary club secretary (and, I think, BBS sysop). Thank you John Novotny for last month's newsletter article. And, thank you to the person (I'm sorry I didn't get your name) who made up a beautiful listing of SPACE's disk of the month programs. We will be making this available to the membership in one form or another.

Missing a Coat? - If you left your coat at the last meeting (check your coat closet, you might be surprised), please call me before the meeting to claim it. You must describe it to me. I won't be bringing it to the meeting without a phone call beforehand. I'm surprised no one called after the last meeting.

Demo's - We will demo the club's new 130XE with DOS 2.5 & ramdisk, Mr. Do, the Print Shop, and a special club disk with NICELIST by yours truly. I will be bringing my printer for demo'ing the last two items. NICELIST is a printer utility for printing listings of BASIC programs. It is a compiled BASIC program that uses graphics dumps on the printer to duplicate the Atari character set. Three compiled (and their BASIC counterparts) versions are included as follows:

- 1) NEC8023A/C & Prowriter
- 2) Epson MX80
- 3) Gemini

NICELIST supports a wide variety of printer options. You must see it to believe it. Cost is \$4.00 from Frank or Bruce Haug. Incidentally, Frank Haug has taken over duties as Disk Librarian.

ANALOG D.O.M.

Bob & Joanne Floyd

I will have Analog Computing disks for issues #31 and 32 for sale to registered Analog subscribers at the meeting. Cost is \$4.00 each. I may have #33 also. Please register your subscription with me if you haven't done so already.

DISK OF THE MONTH
AND OTHER TOPICS FROM THE LIBRARY

The disk/tape of the month contains programs from the club library. The tapes can be purchased for \$3 and the disks for \$4. The group makes a small profit from the sale of these which go toward supporting some of the things the group does. Each disk/tape usually contains eight to twelve programs. Most of these programs are games with some utilities and demos mixed in. The programs are obtained from trading with other user groups and from submissions from local members. Members who submit a program will receive a disk or tape of the month free! What a deal!!

Well, before I give you the programs on this disk, I'd like to tell you some things about the disk menu. 'D:SPMENU' can show up to 24 programs. To save space it doesn't show SYS, DAT, or OBJ files. If you copy the menu off the SPACE disk for your own disks, copy the 'D:AUTORUN.SYS' file too. (Some older DOM's have different autorun files and different menus so make sure your autorun will run the menu you are using.) You should name your binary load files with '.COM' for an extender. If you type an 'R' for the selection prompt, the program will re-run itself. A 'D' will load DOS. Going to DOS will erase the program in memory. And finally, during the 'title page', hitting the space bar or pressing the trigger will speed you through it.

If you want a disk directory from BASIC, you can use this two line program.

```
1 OPEN #1,6,0,"D:*.*)"
2 GET #1,A: ? CHR$(A);:GOTO 2
```

When you RUN this, the disk will whirl and the names of the programs on the disk as well as their length and the free sectors on the disk will appear on the screen. You will also see an ERROR 136. Don't worry, this just means that you hit the end of the directory. DON'T FORGET TO ERASE lines 1 & 2 BEFORE YOU SAVE YOUR PROGRAM OR TRY TO RUN IT, THE PROGRAM WILL ERROR ON LINE 2.

--THIS MONTH'S PROGRAMS--

1. HIGHLITE.BAS - When you run this program it will load a machine language subroutine into page 6. The lines above the cursor will be black, the ones below will have the same blue background, and the line it is on will be dark blue.

2. RESQSUB.COM - In this SPACE member-written program, you, the pilot of a submarine must rescue people from their pods and bring them back to your launching pad.

You can read the rest of the instructions on the disk, in the game itself.

3. DEFEND.COM - Shoot aliens to accumulate a high score, but don't let them touch you.

4. ABCTRAIN.BAS - This program is an educational game in which you move the joystick left & right to select the proper letters of the train cars and press the trigger to put it on the truck.

5. POKERSOL.BAS - Poker with a twist, you place 25 cards on a 5x5 grid aiming for as high a score as you can get both horizontally and vertically.

6. DISKFIX.COM - A disk utility that helps you retrieve deleted files, if you haven't written out there since it was deleted. It also has other functions.

7. FID.COM - This program will allow you to copy multiple files and LOCK or UNLOCK selected files. It also has other disk related functions.

For those of you who are new to the D.O.M. here are some operating hints. After you are finished with one program and you want to bring in another, simply hit the break key, and then type in RUN "D:SPMENU". This will return you to the space menu.

Not all of the programs on some of the older D.O.M.'s work. I would like someone to make a list of these, and then ask for a volunteer to fix the problems and then I will make up a new disk.

June, 1985 Meeting Minutes
by Steve Pauley

The last meeting was held on June 14, 1985 at 7:30 p.m. Bob Floyd, the President, started the meeting with news about the special 520ST deal from Atari. Steve Pauley, of Computer Food, explained his better deal on 520STs to the club. Steve also talked about new software planned for the 520ST that will be available when the machine is shipped or shortly after. Atari is planning to ship the new 520ST about July 8th. The 260ST will be shipped sometime in September according to an Atari press release at CES.

Yes, the 130ST is dead and has been replaced by a non-expandable 260ST(256K) machine. This machine is slated to be sold at discount retail stores like K-Mart & etc.

The 520ST will come in two versions. The monochrome version will include the computer (512K), TOS (Trimel Operating System) by

Digital Research, GEM (Graphics Environment Manager) also by Digital Research, Monochrome Monitor, 3 1/2" Disk Drive, LOGO language and BASIC language. GEM and TOS built into ROM chips will be available this fall to all owners as a maximum \$20 upgrade. The color version is the same except a color RGB monitor is in the package instead of the monochrome one. List price for the color version is \$999 and the monochrome version lists for \$799.

The new CD-ROM player that was announced at CES was talked about by Bob. Bob also covered other news from CES about new Lucasfilm games, Batteries Included's 80 col. card, AtariWriter+, Silent Butler and several programs from Haba Systems.

Bruce Haug talked about the DOM, which included a Disassembler and Skiing game with editor (Bruce's favorite). Bruce was not sure that the previously announced 'C' program was public domain.

The club has 100+ paid members now and a bank balance of \$852. The BBS has been up and down and the club is looking for a new home for the board so that it can be better maintained.

Various members told of hot deals around town which shows, 'you should keep your eyes open for stores dumping Atari stuff' (high tech word).

The meeting was concluded with demo of these programs: Skiing II, Bare Essentials, Milleped, Moon Patrol and Tapper.



CHRIS CRAWFORD
ASSEMBLY LANGUAGE COURSE
FOR WORLDWIDE USERS NETWORK
LECTURE ONE - 5/28/85

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, 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.

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 is composed of protons 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. 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 Lru.

PROGRAMMING A MICROPROCESSOR

Machine code is nothing more than a bunch of numbers that mean something to the CPU. It's hard to work with pure 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:

Machine Code	Assembly Language
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 possible-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 three instructions on the 6502 are:

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)

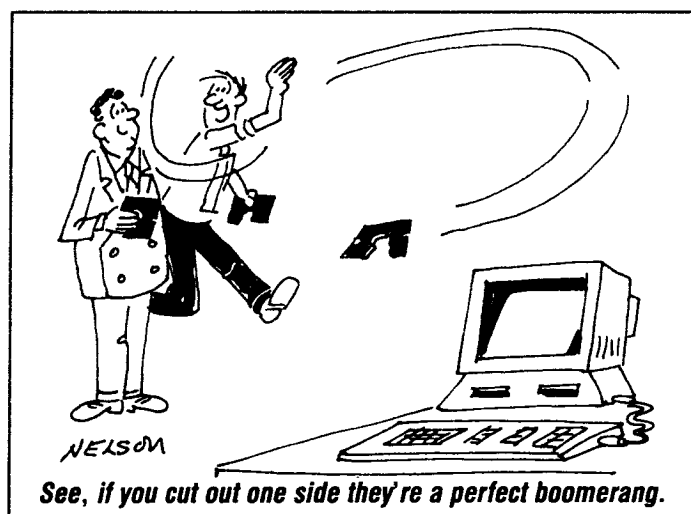
This command will store the contents of the accumulator into the RAM location whose address is specified in the command. It is just like the first command, except that the direction of data motion is reversed. The LDA command is like a read, which the STA is like a write.

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:

```
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.





A DIATRIBE ON THE WAR AGAINST PIRACY

by Hugh McLean
Sysop, BUNKY'S BOARD
(313) 541-5586

DISCLAIMER-This editorial is not in any way representative of MACE, any other organization, club, or ethical group, and does not represent any position of local BBS sysops or user groups. The arguments and suppositions contained in this article are entirely the opinion of the author.

The word "pirate" is a rather nasty sounding word, indicative of a rogue of the high seas who pillaged, looted, raped and stole anything that he could find... plundering gold, jewels, women, ships and cities, and in general acting like an uncouth, uncivilized, ruthless barbarian. The word "pirate" is also used to describe someone who engages in unlawful copying, selling, and distribution of literary material or other copyrighted work. The word as applied to computers reflects the idea that the swapping of copyrighted programs is akin to the pillaging of the Queen's private ship; that it is highly illegal and deserving of at least the death penalty. And indeed, it may be illegal, as it does appear to violate the copyright laws of our land. But whether the deed is intrinsically illegal or immoral is not the issue for me.

I hear the bellicose voice of self-esteeming justice declaring the alleged purism of the righteous, that "trading games" is strictly verboten, and that you are only hurting yourselves by being part of this clandestine movement of pilfering programs. As a stockholder of Warner Communications, I can appreciate the fact that profits are considerably reduced by people who do not purchase programs at the local computer store,

or do not play the games at the local video arcade. It should be against the law to play a computer game at home that you didn't pay at least \$50 for. After all, you have to consider the poor bit twiddlers who make their livings creating games for the gullible, inane public to play.

I'm prejudiced, I guess. I don't like to see "piracy" going on either, but the pirates that I refer to are the people who think that the American public is some kind of "Golden Goose" with a golden egg, just ready to be plucked... and the bigger the egg, the better. They don't mind slitting throats, corporate-wise. The recent demise of Texas Instrument Computers I believe was a result of undercutting computer prices to kill the competition while making it up by means of overpricing software. So what if Atari, Coleco, etc. make a windfall profit? The good ole goose can afford the added cost. Anyway, she is too stupid to know differently. Charge as much as you can while the market is there; the customer will pay, and forget what he paid when the "new models" come out and you reduce the price. Gouge while you can, and convince the public that it is really getting a deal.

This sysop says that the real pirates are the ones who rip you off in the store, by charging five times the price the program is really worth. You can't convince me that if the price of software were equivalent to the price of a good book or record, the problem of pirating would even exist. Most people would go to the store and buy programs for \$10-15, but when the price is inflated to \$50, they go elsewhere.

I don't go along with ATARI's philosophy on selling software, but I think the real pirates are in the corporate offices of the software companies. Do you really think that a bit twiddler is starving because his software is being traded? Does it take less time to write a book or compose a piece of music than it does to write a game? You can buy a book which took one year to write for under \$5, and you can buy an album or tape for under \$10. If the price of software were where it should be, relative to everything else, the volume of sales would go up, the bit twiddler wouldn't see any difference in his income, and illicit swapping of programs would be diminished considerably.

MENU page #1

```

5 DIM A$(17),B$(560),C$(15),D$(16),E$(
20),F$(11),G$(14),H$(14),I$(1),J$(27)
10 OPEN #2,6,0,"D:*.*":TRAP 30:POKE 82
,0:POKE 83,39:GRAPHICS 0:POKE 710,146:
POKE 752,1:POKE 53774,64:POKE 16,64:XX
=1:AR=9
20 FOR X=1 TO 510 STEP 17:INPUT #2,A$:
B$(X)=A$:NEXT X
30 ? "K",
";:FOR I=1 TO 21:POKE 85,0:?
"|";:POKE 85,18:? "|" |:POKE 85,39:?
"|";
40 NEXT I:? "
";
50 TRAP 60:X=1:FOR I=1 TO 21:POSITION
1,I:? B$(X,X+16):X=X+17:NEXT I:FOR I=1
TO 13:POSITION 22,I:? B$(X,X+16):X=X+
17:NEXT I
60 POSITION 21,12:? "
";:POSITION 22,13:? "C-Catalog":POSIT
TION 22,14:? "D-Delete"
65 POSITION 22,15:? "F-Format":POSITIO
N 22,20:? "P-Print menu"
70 POSITION 22,16:? "L-Lock":POSITION
22,17:? "U-Unlock":POSITION 22,18:? "R
-Rename":POSITION 22,21:? "Q-Quit"
75 POSITION 0,0:? "0 ";A$:GOSUB 600
78 POSITION 18,0:? " SELECT WITH ARRO
WS "
79 POKE 764,255
80 POSITION 22,20:? "SELECT-Type-";:IF
XX=1 THEN ? "LOAD "
90 IF XX=2 THEN ? "ENTER"
100 IF XX=3 THEN ? "BLOAD"
105 PEK=PEEK(764):TRAP 79
110 IF PEEK(53279)=5 THEN XX=XX+1:IF X
X=4 THEN XX=1
120 IF PEK=255 THEN 78
122 POSITION 18,0:? "PRESS RETURN TO B
EGIN"
130 IF PEK=14 THEN GOSUB 800:AR=AR-1:G
OSUB 600:GOTO 79
134 IF PEK=6 THEN GOSUB 800:AR=AR-21:G
OSUB 600
136 IF PEK=7 THEN GOSUB 800:AR=AR+21:G
OSUB 600
140 IF PEK=15 THEN GOSUB 800:AR=AR+1:G
OSUB 600:GOTO 79
145 IF PEK=10 THEN GOSUB 500:GOTO 400
147 IF PEK=12 THEN 700
150 IF PEK=18 THEN RUN
160 IF PEK=58 OR PEK=56 OR PEK=0 OR PE
K=11 OR PEK=40 THEN GOSUB 500:GOTO 200
170 IF PEK=47 THEN GOSUB 500:GOTO 350
180 GOTO 79
200 TRAP 740:IF PEK=58 THEN IO=33:F$="
DELETE FILE"
210 IF PEK=0 THEN IO=35:F$="LOCK FILE"
215 IF PEK=11 THEN IO=36:F$="UNLOCK FI
LE"
222 POKE 764,255:IF PEK=56 OR PEK=40 T
HEN 240
225 POSITION 22,14:? F$:POSITION 22,15
:? "Enter file name"
230 POSITION 22,16:INPUT G$:XIO IO,#1,
0,0,G$
235 RUN
240 IF PEK=40 THEN IO=32:F$="RENAME FI
LE":GOTO 260
245 IO=254:F$="FORMAT DISK"
250 POSITION 22,14:? F$:POSITION 22,15
:? "Are you":POSITION 22,16:? " sure (
Y/N)":INPUT I$:IF I$(>)"Y" THEN 30
255 XIO IO,#1,0,0,"D:"
260 POKE 764,255:POSITION 22,14:? F$:P
OSITION 22,15:? "Enter old name":POSIT
ION 22,16:INPUT G$
262 POSITION 22,15:? "Enter new name":
POSITION 22,16:INPUT H$
265 J$(1,14)=G$(1,14):J$(15)="," :J$(16
,30)=H$(1,12)
270 XIO IO,#1,0,0,J$
350 POSITION 22,16:? "Press RETURN"
360 POSITION 22,17:? "To quit, or"
365 POSITION 22,18:? "SYSTEM RESET"
370 POSITION 22,19:? "To reboot."
380 POKE 580,1:POKE 764,255
390 IF PEEK(764)=12 THEN POKE 580,0:PO
SITION 0,23:POKE 764,255:POKE 752,0
D
395 GOTO 390
400 POSITION 22,14:? "Disk title":POSIT
TION 22,15:? "(16 letter max)":POSITIO
N 22,16:POKE 764,255:TRAP 430:INPUT D$
410 LPRINT D$:LPRINT :ZZ=1
420 LPRINT B$(ZZ,ZZ+16):ZZ=ZZ+17:GOTO
420
430 IF PEEK(195)=138 THEN POSITION 22,
19:? "CHECK PRINTER.":FOR I=1 TO 100:
NEXT I:GOTO 30
435 LPRINT A$:LPRINT :GOTO 30
500 Z=14
510 POSITION 22,Z:? "
":Z=Z+1:IF Z=23 THEN RETURN
520 GOTO 510
600 IF AR<1 THEN AR=1:RETURN
605 IF AR>32 THEN AR=32
610 IF AR<22 THEN POSITION 19,AR+1:? "
←-"
615 IF AR>22 THEN POSITION 19,AR-20:?
"-→"
617 TRAP 640
620 C$(1,2)="D:" :C$(3,10)=B$((AR-1)*17
+3,(AR-1)*17+10):C$(11)="," :C$(12,1
)=B$((AR-1)*17+11,(AR-1)*17+13)
625 IF C$(3,5)="DOS" THEN XX=5
630 RETURN
640 POP :GOTO 79
700 TRAP 740

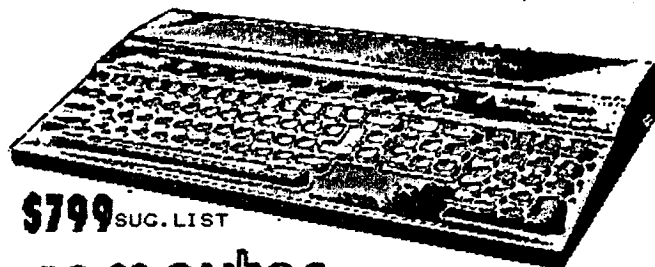
```

```

705 IF XX=1 THEN RUN C$
707 IF XX=5 THEN DOS
710 IF XX=2 THEN POSITION 0,23: ? : ? : ?
3 "NEW": ? : ? : ? "ENTER ";CHR$(34);C$:PO
KE 764,255:POKE 752,0:POSITION 0,16:EN
D
720 E$=C$:BUFHI=INT(ADR(E$)/256):BUFLO
=ADR(E$)-BUFHI*256:POKE 852,BUFLO:POKE
853,BUFHI
730 X=USR(ADR("hlu="))
740 GOSUB 500:POSITION 22,15: ? "ERROR
-";PEEK(195):FOR I=1 TO 100:NEXT I:GOT
O 30
800 FOR I=2 TO 22:POSITION 19,I: ? " "
:NEXT I:RETURN
    
```

This is a sample print-out from Bob Floyd's NiceList program. The program is a menu program written by me. It does more than any other menu program I know of (ENTER's, LOADS BINARY FILES, plus all BASIC-available DOS functions. If you find any errors, fix them and write about it!

Atari 520ST



\$799 SUG. LIST

computer
FOOD

PRICE

\$680

- Monochrome Monitor
- 360K Drive
- TOS & GEM
- BASIC
- LOGO

Color RGB Version - \$850

**Call 560-2917 for
more information**

S.P.A.C.E. Membership Application

Name _____ Phone _____ Date _____

Address _____ City _____ State _____ Zip _____

Equipment/System-Used _____

Area of interest _____

Special skills, interest, info which might be shared. _____

Optional
information: Occupation _____ Employer _____

Dues: \$10.00 per year - please inclose check with application.

Mail application and dues to:
St. Paul Atari Computer Enthusiasts
748 Amber Dr., Shoreview, MN 55112

WANT ADS

It is possible for members to place 'Want Ads' in this newsletter. The ads may be for selling used hardware, used software, tutoring services, or just about anything that has to do with Atari. The rates are as follows:

6 Lines \$1.25 216 Letters
1 Line .25 36 Letters

The following is a list of advertising rates for vendors or individual members.

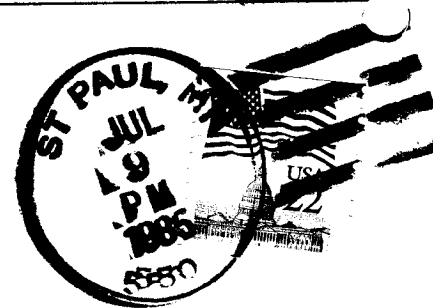
Full page \$18.00 7-1/2" X 10"
Half page 10.00 3-5/8" X 10"
Half page 10.00 7-1/2" X 5"
1/4 page 5.50 3-5/8" X 5"

All advertisements must be paid for when they are submitted. Deadline for ad placement is the last Friday of each month. To place ad or for more info, call the editor.

Published by the St. Paul Atari Computer Enthusiasts (SPACE), an independent organization with no business affiliation with Atari Incorporated. Permission is granted to any similar organization with which SPACE exchanges newsletters to reprint material in this newsletter. We do, however, ask that credit be to the authors and to SPACE. Opinions expressed are those of the article authors and do not necessarily reflect the views of SPACE, club officers, members, or Atari Inc.

Bob Floyd . . . President	487-2627	Jon Nelson . . . Editor	484-9027
Bruce Haug . . . Vice-Pres	774-6226	Max Feuer . . . Treasurer	483-3895
Frank Haug . . . Disk Lib.	774-6226	Bruce Haug . . . Cass Lib.	774-6226
Jim Schulz . . . Paper/Ed. Lib.	537-5442	Steve Pauley . . . Secretary	560-2917
SPACE/TAIG Bulletin Board . . . 544-9058			

St. Paul ATARI Computer Enthusiasts
2589 Fisk St.
Roseville, MN 55113



J. P. Scheib
3944 24th Av. S.
Minneapolis, Mn. 55406

Next SPACE Meeting:
Friday, July 12, 1985
7:30 P.M.

U of M St. Paul campus
Classroom Office Bldg. rooms B-35
B-36
St. Paul, Minnesota