creative computing

January 1981
vol 7, no 1
$2.50

the #1 magazine of computer applications and software

Graphics and Animation

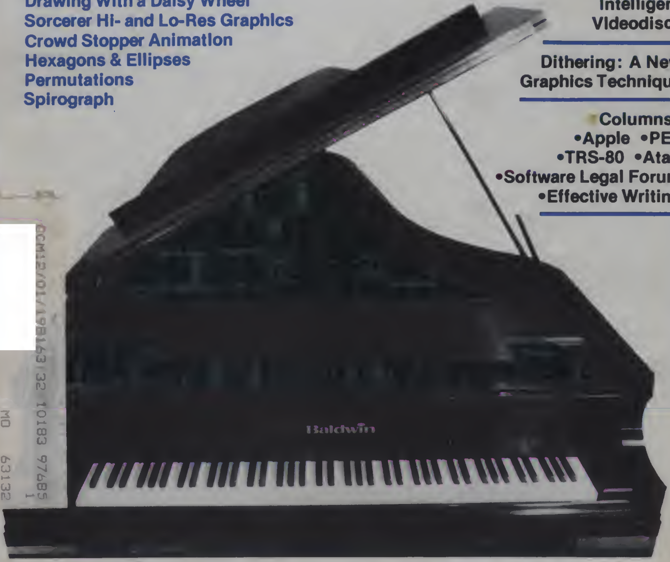
- Printing Without Quotes
- Apple-Sketch
- Apple Lo-Res Shape Tables
- Apple Hi-Res Text
- Drawing With a Daisy Wheel
- Sorcerer Hi- and Lo-Res Graphics
- Crowd Stopper Animation
- Hexagons & Ellipses
- Permutations
- Spirograph

Evaluations:
- TRS-80 Color Computer
- Apple Graphics Tablet
- Atari Music Composer
- Casheab Music Synthesizer
- New PETs

Intelligent Videodiscs

Dithering: A New Graphics Technique

Columns:
- Apple
- PET
- TRS-80
- Atari
- Software Legal Forum
- Effective Writing



The Perfect Fit

The Micromodem II data communications system and the Apple II* computer. What better combination to maximize the capabilities of your personal computer!

This popular direct connect modem can transmit data between an Apple II and another Apple II, a terminal, another microcomputer, minicomputer or even a large time-sharing computer anywhere in North America. The Micromodem II has unique automatic dialing and answer capabilities which further increases the communications possibilities between the Apple II and another computer or terminal.

You can send and/or receive messages or data when you are out of your office, home or out of town. Your branch business locations can communicate with each other regarding inventory and other matters over the phone. Or you can communicate with friends across the country. And you can access information utilities like the SOURCE for various business and personal applications.

The Micromodem II consists of two parts. One part includes the printed circuit board which holds the Micromodem II, ROM firmware and the serial interface. The board plugs directly into the Apple II providing all the functions of a serial interface card plus programmable auto dialing and auto answer capabilities. The on-board ROM firmware enables the Micromodem II to operate in any of three modes to perform different tasks-terminal mode, remote console and program control mode.

The other part of the Micromodem II datacomm system is a Microcoupler which connects the Micromodem board and Apple II to a telephone line. The Microcoupler gets a dial tone, dials numbers, answers the phone and hangs up when a transmission is over. There are none of the losses or distortions associated with acoustic couplers. The Microcoupler is compatible with any North American standard telephone lines and is FCC-approved for direct connection in the U.S. It works with standard dial phone service or Touch-tone service.

The Micromodem II is completely compatible with Bell 103-type modems. Full and half-duplex operating modes are available as well as speed selectable transmission rates of 110 and 300 bps.

Why not increase your Apple II's capabilities by outfitting it with the sophisticated Micromodem II data communications system? The Micromodem II is available at retail computer stores nationwide. For the store nearest you, call or write:

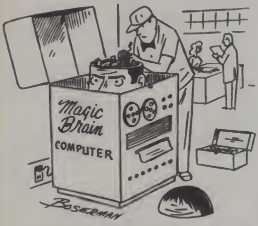
Hayes™

Hayes Microcomputer Products Inc.
5835 Peachtree Corners East, Norcross, Georgia 30092 (404) 449-8791

™ Micromodem II is a trademark of Hayes Microcomputer Products, Inc.
* Apple II is a registered trademark of Apple Computer Inc.
The Micromodem II can also be used with the Bell & Howell computer.

CIRCLE 153 ON READER SERVICE CARD

creative computing



Editorial Features 1981

February:
Artificial Intelligence. Interactions in the AI community.

March:
Educational applications. Winter CES coverage.

April:
Networks, large and small. Data transmission and communications for personal and larger computers.

May:
Investment analysis and financial decision making for individuals and business firms.

June:
Computer graphics, art and movie-making techniques.

July:
Word processing, text editing, printers.

August:
Simulations and models. Simulating both continuous and discontinuous processes. Data Bases. NCC coverage.

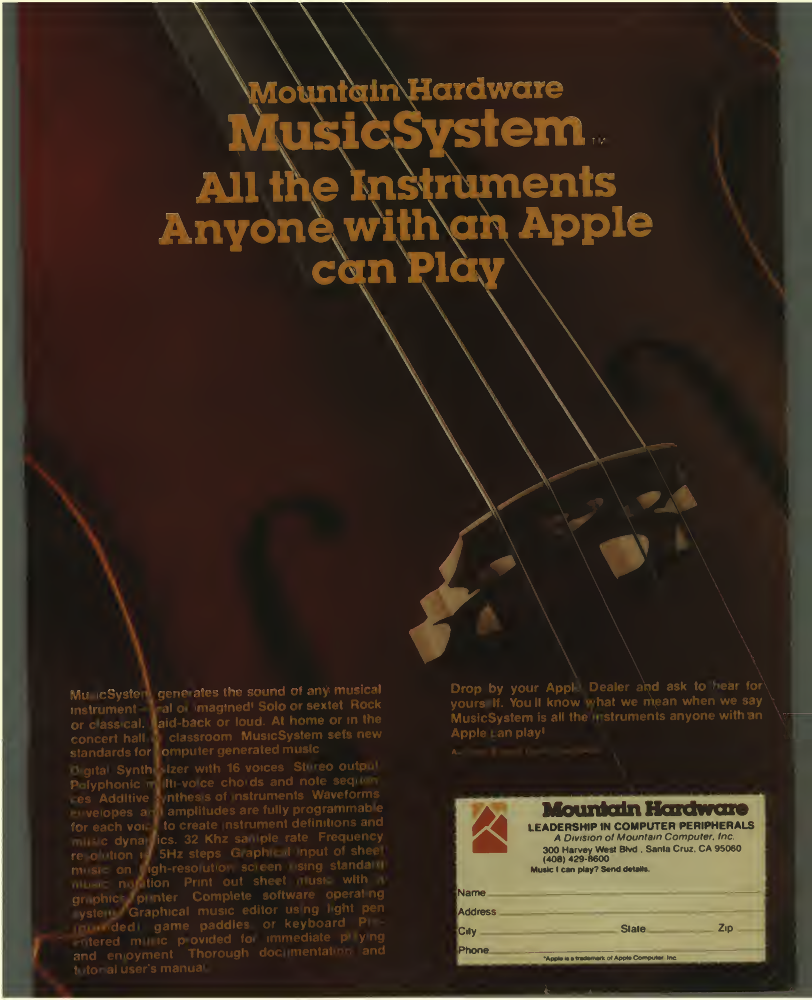
September:
Educational applications. Summer CES coverage.

October:
Getting started with a personal computer. Techniques for advanced users too.

November:
Problem solving, programming techniques. Electronic toys and games.

December:
Buying guide to personal computers and consumer electronic products.

#1 in Software and Applications



Mountain Hardware
MusicSystem™
All the Instruments Anyone with an Apple can Play
MusicSystem generates the sound of any musical instrument—real or imagined! Solo or sextet. Rock or classical. Laid-back or loud. At home or in the concert hall or classroom. MusicSystem sets new standards for computer generated music.
Digital Synthesizer with 16 voices. Stereo output. Polyphonic multi-voice chords and note sequences. Additive synthesis of instruments. Waveforms, envelopes and amplitudes are fully programmable for each voice to create instrument definitions and music dynamics. 32 Khz sample rate. Frequency resolution in .5Hz steps. Graphical input of sheet music on high-resolution screen using standard music notation. Print out sheet music with a graphics printer. Complete software operating system. Graphical music editor using light pen (included), game paddles, or keyboard. Pre-entered music provided for immediate playing and enjoyment. Thorough documentation and tutorial user's manual.
Drop by your Apple Dealer and ask to hear for yourself. You'll know what we mean when we say MusicSystem is all the instruments anyone with an Apple can play!
Mountain Hardware
LEADERSHIP IN COMPUTER PERIPHERALS
A Division of Mountain Computer, Inc.
300 Harvey West Blvd., Santa Cruz, CA 95060
(408) 429-8600
Music I can play? Send details.
Name
Address
City
State
Zip
Phone
*Apple is a trademark of Apple Computer, Inc.

In This Issue

applications - games

60 **Crowd Stopper** .... *Ross*
Tips for effective animation

64 **Artist/Computer Dialogue** .... *Demio*
One man's approach to graphics

66 **Permutations** .... *Ridell*
Another man's approach to graphics

74 **An Atari Library of Sound** .... *Kruse*
Bells and Whistles

80 **Many Faces** .... *Schwandt*
Students make their Marx

86 **Hexagons and Ellipses** .... *Speer*
Shapely programs

88 **Grapefruit Delight** .... *Jacobson*
Tektronix meets Trigonometry

92 **Spirograph** .... *Sietz*
Toying around

94 **Hi-res for the Sorcerer** .... *Gass*
Doubling the resolution

98 **Lo-Res for the Sorcerer** .... *Stuckmeyer*
An aid to drawing

102 **Sorcerer Meets Paper Tiger** .... *Stuckmeyer*
Hardcopy the easy way

110 **Apple-Sketch** .... *Miller*
Poor man's graphics

120 **Apple Lo-Res Shape Tables** .... *Lubar*
For rapid animation

126 **Hi-Res Text for the Apple** .... *Hitchcock*
Letter those charts and pictures

130 **Lit'l Red Bug** .... *Bishop*
Take the VW for a spin

132 **The Digital Couch** .... *Tunbo*
Shrinking to new depths

136 **Printer Plotting** .... *Kalush*
Drawing with a daisy wheel

142 **How to Solve It** .... *Piele*
Coin tossing and bug-free programs

JANUARY 1981 **VOLUME 7, NUMBER 1**

Creative Computing magazine is published monthly by Creative Computing, P.O. Box 789-M, Morristown, NJ 07960. (Editorial office 39 East Hanover Ave., Morris Plains, NJ 07950 Phone: (201) 540-0445.)

Domestic Subscriptions: 12 issues $20; 24 issues $37; 36 issues $53. Send subscription orders or change of address (P.O. Form 3575) to Creative Computing P.O. Box 789-M, Morristown, NJ 07960. Call 800-631-8112 toll-free (in New Jersey call 201-540-0445) to order a subscription (to be charged only to a bank card)

Controlled Circulation paid at Richmond, VA 23228

evaluations & profiles

14 **Soft Centered** .... *Lubar*
Death and taxes

16 **TRS-80 Color Computer** .... *Garon*
A low-cost color graphics system

24 **No PET Peeves** .... *Fee*
New computers from Commodore

26 **Atari Music Composer** .... *Nasman*
Beethoven on a chip

28 **Apple Graphics Tablet** .... *Sternecker*
For the drawing room

30 **Casheab Music Synthesizer** .... *Bondy*
A sound S-100 set up

36 **Educational Software and Books** .... *Lubar*
New entries in the field

articles

40 **Sights and Sounds** .... *Lubar*
Two days of sensory gluttony

44 **Siggraph** .... *Wallace*
The latest graphics hardware

50 **Dawn of the Disc** .... *Lubar*
Will video discs make a difference?

56 **Sesame Place** .... *Staples*
Opening day at the play park

108 **Dithering Heights** .... *Recters*
Video images through the Apple

departments

6 **Et Cetera** .... *Et al*

8 **Input/Output** .... *Readers*

68 **Appallogy** .... *Nelson*

152 **Retail Roster** ....

156 **Personal Electronic Transactions** .... *Yob*
Printer list, ROM, big Keyboard and more Hangmath

164 **TRS-80 Strings** .... *Gray*
String packing, new software, a robot van vanishes

170 **Apple Cart** .... *Carpenter*
DOS 3.3, the Z-80 card, and an Apple bug

176 **Outpost: Atari** .... *Blank*
Missile graphic mysteries revealed

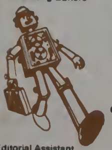
180 **Effective Writing** .... *Weiss*
Say a little more

182 **Software Legal Forum** .... *Novick*
The controversy continues

184 **Puzzles & Problems** .... *Townsend*

186 **Book Reviews** .... *Gray*

192 **Index to Advertisers** ....



Advertising Sales

Advertising Coordinator
Renee Christman
Creative Computing
P.O. Box 789-M
Morristown, NJ 07960
(201) 540-9168

Western States, Texas
Jules E. Thompson, Inc.
1290 Howard Ave., Suite 303
Burlingame, CA 94010
(415) 348-8222

Southern California
Jules E. Thompson, Inc.
2560 Via Tejon
Palos Verdes Estates, CA 90274
(213) 378-8361

Mid-Atlantic, Northeast
CEL Associates, Inc.
27 Adams Street
Braintree, MA 02184
(617) 848-9306

Midwest
Ted Rickard
435 Locust Rd.
Wilmette, IL 60091
(312) 251-2541

New York Metropolitan Area
Nelson & Miller Associates, Inc.
55 Scenic Dr.
Hastings-on-Hudson, NY 10706
(914) 478-0491

Southeast
Paul McGinnis Co.
60 East 42nd St.
New York, NY 10017
(212) 490-1021

Responsibility

Creative Computing will not be responsible for the return of unsolicited manuscripts, cassettes, floppy disks, program listings, etc. not submitted with a self-addressed, stamped envelope.

OK to Reprint

Material in Creative Computing may be reprinted without permission by school and college publications, personal computing club newsletters, and non-profit publications. Only original material may be reprinted; that is, you may not reprint a reprint. Also, each reprint must carry the following notice on the first page of the reprint in 7-point or larger type (you may cut out and use this notice if you wish):

Please send us two copies of any publication that carries reprinted material. Send to attention: David Ahl.

Microform

Creative Computing is available on permanent record microfilm. For complete information contact University Microfilms International, Dept. F.A., 300 North Zeeb Road, Ann Arbor, MI 48106 or 18 Bedford Road, London WC1R 4EJ, England.

Foreign Customers

Foreign subscribers in countries listed below may elect to subscribe with our local agents using local currency. Of course, subscriptions may also be entered directly to Creative Computing (USA) in U.S. dollars. (bank draft or American Express card). All foreign subscriptions must be prepaid.

Many foreign agents stock Creative Computing magazines, books, and software. However, please inquire directly to the agent before placing an order. Again, all Creative Computing products may be ordered direct from the USA — be sure to allow for foreign shipping and handling.

CANADA	Surface	Air
1-year	C $26	n/a
2-year	54	
3-year	78	

Micron Distrib.
409 Queen St. W.
Toronto, OT M5V 2A5, Canada

ENGLAND	£	£
1-year	10.00	19.00
2-year	19.50	37.25
3-year	28.50	55.00

CREATIVE COMPUTING
Attn: Hazel Gordon
27 Andrew Close
Stoke Golding, Nuneaton CV13 6EL
England

FRANCE	F	F
1-year	96	183
2-year	186	358
3-year	273	530

SYBEX EUROPE
14/16 Rue Planchet
75020 Paris, France

SWEDEN	kr	kr
1-year	100	196
2-year	193	368
3-year	280	544

HOBBY DATA
Attn: Jan Nilsson
Fack
S-200 12 Malmo 2, Sweden

GERMANY	dm	dm
1-year	42	78
2-year	80	152
3-year	118	226

HOFACKER-VERLAG
Ing. W. Hofacker
8 Munchen 75
Postfach 437, West Germany

HOLLAND, BELGIUM		f
1-year		106
2-year		210
3-year		300

2XF COMPUTERCOLLECTIEF
Attn: F. de Vreeze
Amstel 312A
1017 AP AMSTERDAM, Holland

AUSTRALIA	$A	$A
1-year	23	47
2-year	44	92
3-year	64	136

ELECTRONIC CONCEPTS PTY., LTD.
Attn: Rudi Hoess
Ground Floor 55 Clarence St.
Sydney, NSW 2000, Australia

JAPAN	Y	Y
1-year	5,700	10,700
2-year	10,900	21,000
3-year	15,900	31,000

ASCII PUBLISHING
Aoyama Building 5F
5-16-1 Minami Aoyama, Minato-Ku
Tokyo 107, Japan

ITALY	IL	IL
1-year	27,500	47,500
2-year	43,500	65,500
3-year	59,000	79,000

ADVEICO S.R.L.
Via Emilia Ovest, 129
43016 San Pancrazio (Parma) Italy
Attn: Giulio Bertellini

PHILIPPINES	P	P
1-year	175	330
2-year	338	650
3-year	490	955

INTEGRATED COMPUTER SYSTEMS, INC.
Suite 205, Limketkai Bldg., Ortigas Ave.
Greenhills P.O. Box 483, San Juan
Metro Manila 3113, Philippines

OTHER COUNTRIES		
1-year	$24	$45
2-year	46	88
3-year	67	130

CREATIVE COMPUTING
P.O. Box 789-M
Morristown, N.J. 07960, USA

et cetera
et cetera

National Computer Camp

This summer youngsters can again sign up for an overnight camp in Moodus, Connecticut, where the main activity will be computers. The camp will be directed by Dr. Michael Zabinski, Professor at Fairfield University.

The 1981 National Computer Camp will feature two one week sessions: 1st session July 19-24; 2nd session July 26-31. The campers, ages 10—17, will enjoy small group instruction and mini and personal computers for ample "hands-on". Dr. Zabinski will be assisted by high school teachers.

The camp is for kids of all levels of experience including no experience whatsoever. In addition to computers, the campers will enjoy the superb recreational facilities of the Grand View Lodge including swimming and tennis.

For further information, contact Michael Zabinski, PH.D., at (203) 795-9069, or write to: Computer Camp, Grand View Lodge, Box 22, Moodus, CT 06469.

TABS-Math Project

The College of Education of The Ohio State University has begun a project to develop and disseminate exemplary curricular materials in which high technology is used to teach basic mathematical skills including problem solving, estimation, computer literacy, etc. Funded by the U.S. Department of Education, the project will collect and evaluate existing educational software for small computers (e.g., Apple, TRS-80, PET, etc.) and select high quality units for inclusion. Other curricular elements will be developed by the project under the direction of Suzanne Damarin, Marlin Languis, and Richard Shumway. The curricula will be field tested and disseminated nationally. Individuals or groups who have developed programs related to mathematics at the upper elementary school level are invited to submit them for possible inclusion for national dissemination. To have materials considered, send a cassette tape of floppy disk together with a printout, machine documentation and any related information to: Dr. Suzanne K. Damarin, TABS Projects, Arps Hall 202-A, 1945 N. High St., Columbus, Ohio 43210. For further information write or call Dr. Damarin at (614) 422-1257.

Corrections

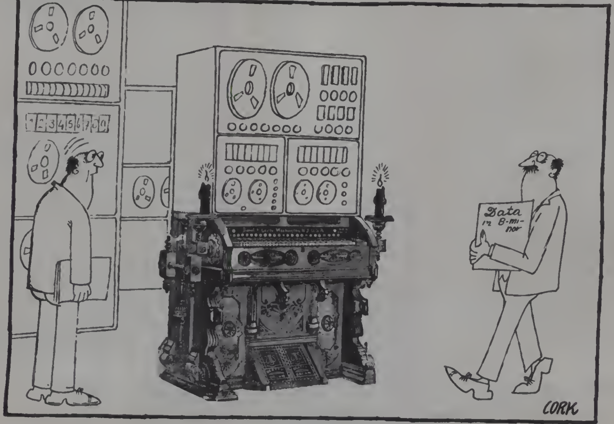
In "The Presidential Campaign" (October '80) on Page 120, Line 21416, the line should read:

```
21416 PRINT"YOU HAVE BEEN FOUND GUILTY AND YOU LOSE"; INT(100/(G-F));:PRINT"PERCENT OF YOUR SUPPORT IN EACH STATE.":GOTO21420
```

The MicroNET number in the October '80 Apple Cart (Page 167) was listed incorrectly. It should be: (614) 457-8600. The per hour charge for the service is $5, not $4.

TSUNAMI

The Sorcerer Users' Newsletter Around Michigan is a free bi-monthly newsletter for Exidy Sorcerer owners which concentrates on advanced applications in a variety of languages. Past issues have included a 7 generation / second Life, a Paper Tiger screen printer program, etc. SASE's and articles are requested but not required. Write: Joseph R. Power, 124 Cedar St #5, E. Lansing, MI 48823, (517) 337-1049.



Input/Output

Re Creation

Dear Editor:

Raudsepp's "New Look at the Creative Process" (August '80) was most stimulating and rich in quotable references. Aaron Copeland's description of the moment of inspiration as a hallucination in split consciousness is similar to a dream state. A person can train oneself to retain a conscious detachment, an observer role, during dreams. If the dream is not too deep, the dream can be directed. For example, the conscious state can try to read some written matter in a dream moving a document in closer for examination. The observed text in this dreamer's experience turns out to be nonsense words. This is quite similar to Copeland's description of events during a creative mood.

Does Princeton Creative Research recognize different types of creativity? Generally, Raudsepp's article seems to address artistic creativity rather than problem solving creativity. The free form, even randomness, of creativity from a dreamlike state may suit the artist, but a scientist or engineer is likely to be creative from a more structured and orderly state. A problem is contemplated, puzzled over; it is turned over in the mind, viewed from different aspects and in abstract. The mind characterizes the problem by asking: "What is the problem? Is it well defined? What are the boundary conditions? What is known as opposed to what is only a presumption?" Then one can modify each assumption or set of assumptions, seeing if a solution presents itself, if the problem has similarities to other problems, or if it has analogs in one's experience.

This process is most creative, especially at the abstract level, but it lacks the randomness of the dream state. Still, answers are elusive and fleeting, hard to recover if not grasped quickly at the conscious level. Sometimes the ideas that come forth are most difficult to verbalize or otherwise make concrete. Sometimes the ideas are lost or are too abstract, in which case associative searches of the mind are appropriate wherein it may pay to write down thoughts that come to mind on the subject.

This type of creativity is very much like searching for a key to a lock, but the description of the lock is incomplete or deceptively inaccurate. This is a problem in pattern matching and it may be akin to those states of mind labeled by the word intuition of déjà-vu. In the end, however, since a solution to a problem is sought, the answer must stand the test of logic and a critical appraisal. The trick is to bring on this final phase of evaluation late in the process, as Raudsepp has correctly observed.

Raudsepp seems contradictory in saying first that "Critical judgement must have recourse to past experience... . Being a past-oriented way of thinking, it is essentially opposed to the novel, the untried and the original." Later, he states, "During the heat of creative forming, the creative individual has to abandon himself entirely to his experience... ." The experience of the past brings fruition to intuition. It supplies the creative process as well as the abstract template for problems and their solutions. Too much experience can block the creative process, however, if one fails to work against the natural reluctance to replow old ground.

Having correctly identified early application of the critical attitude as defeating to creativity, Raudsepp then attacks critical judgement itself and not its premature application as the issue. In the problem solving situation, critical judgement is essential. When the time is right, creative engineers will themselves apply this critical process and they will seek it from others whom they know can be constructive.

To deprecate the critical attitude is to misunderstand the creative process in engineering and science. This misunderstanding is compounded by Dr. Goshen's stereotyping of engineers, which is as accurate and as informative as any sterotyping. The cerebral cortex is as developed in engineers as it is in psychologists; there is as much variety in personality among engineers as there is in any other profession or trade. Dr. Goshen's stereotypes are certainly not the "highs," and if his comments are substantiated by his observations, then he needs to be guided to a more representative group of subjects.

One last comment: there surely are many techniques in the creative process and there may be more forms of creativity other than artistic and problem solving. Brainstorming is one such technique and perhaps Madison Avenue creativity is an example of the second type. Brainstorming has its advocates and it might fit the advertising business, but it has no place in engineering or science.

J.A. Glassman
Chief Scientist
Hughes Aircraft Company
Missile Systems Group
Canoga Park, CA 91304

Edison had over 1,800 patents in his name, but you can be just as inventive with an Apple.

Apple is the company with the brightest ideas in hardware and software *and* the best support — so you can be as creative with a personal computer system as Edison was with the incandescent bulb.

How Apple grows with you.

With Apple's reliable product family, the possibilities of creating your own system are endless. Have expansion capabilities of 4 or 8 accessory slots with your choice of system.

Expand memory to 64K bytes or 128K bytes. Add an A to D conversion board. Plug into time sharing, news and electronic mail services. Use an IEEE 488 bus to monitor lab instruments. Add 4 or 6 disk drives — the 5¼", 143K bytes, high-speed, low-cost drive that's the most popular on the market.

Apple speaks many languages.

Since more than 100 companies create software for Apple, you'll have the most extensive library in the personal computer world. Want to write your own programs? Apple is fluent in BASIC, Pascal, FORTRAN, PILOT and 6502 assembly language.

There's even a series of utility programs called the DOS Tool Kit that not only lets you design high-resolution graphic displays, but lets you work wonders with creative animation.

More illuminating experiences in store.

You won't want to miss all the Apple products being introduced at your computer store all the time. Don't let history pass you by. Visit your nearest Apple dealer or call **800-538-9696**. In California, **800-662-9238**. Or write: Apple Computer, 10260 Bandley Drive, Cupertino, CA 95014.

apple computer inc.

Paper Training

```
READY.

 10 REM *** IF YOU'RE A HUMAN, START AT LINE 30
 20 GOTO 370
 30 DEAR CC
 40
 50 IN RESPONSE TO THE LETTER TO YOU FROM MR. CUSHING THAT APPEARED
 60 IN THE SEPTEMBER 80 LETTERS SECTION, THE PET COMPUTER WILL SUPPORT
 70 INDENTING THE BASIC COMMANDS WITHOUT THAT COLON STUFF.
 80    YOU CAN INDENT LINES, SUCH AS THIS ONE, OR YOU CAN HAVE BLANK LINES,
 90 SUCH AS LINE 40.
100
110               THE WAY TO DO THIS IS THIS
120       FOR 'BLANK' LINES, PUT LINE NUMBER, A SHIFTED GRAPHIC SYMBOL OR
130  A LOWER CASE LETTER, A COUPLE OF SPACES, AND ANOTHER SYMBOL.
140
150       FOR A INDENTED LINE, PUT THE LINE NUMBER, A SHIFTED SYMBOL, AND THEN
160      SPACE OVER AS FAR AS YOU WISH.
170  THIS DOESN'T EFFECT THE RUNNING OF A PROGRAM, AS CAN BE SEEN BY
180  RUNNING THE PROGRAM AT THE END OF THIS LISTING.
190
200 USING THE TOOLKIT TO RENUMBER YOUR PROGRAM DOESN'T CHANGE YOUR
210 FORMATTING.  EDITING A EXISTING INDENTED LINE WILL WIPE OUT THE
220 FORMATTING UNLESS YOU PUT BACK THE SYMBOLS.
230
240    THE BEST WAY TO FORMAT A PROGRAM IS TO WRITE THE PROGRAM NORMALLY,
250 USE THE 'INST' KEY TO MOVE THE LINE OVER, AND THEN INSERT THE SYMBOL.
260
270    EASY AS PIE!
280
290    AND IN RESPONSE TO ANOTHER LETTER FROM MR. LETH, THE 03 ROM FOR
300 THE 2022 PRINTER INDEED DID HAVE PROBLEMS.  HOWEVER COMMODORE'S 04
310 ROM, WHICH WAS SUPPOSSED TO CORRECT THESE PROBLEMS, DIDN'T WORK
320 EITHER!  IN FACT, IT CAUSED MORE PROBLEMS THAN IT SOLVED.  COMMODORE
330 HAS GONE BACK TO THE OLD 03 ROM UNTILL A NEW ONE CAN BE DESIGNED.
340    I'M STILL WAITING FOR MY NEW ROM.  (THE 05)
350
360
370 OPEN 4,4
380 FOR I=1 TO 5
390    FOR J=0 TO 4
400      PRINT#4,I+J;,
410    NEXT J
420   PRINT#4
430 NEXTI
440 CLOSE4 END
450
460
470
480                              DAVID M. CONLEY
490                              10571 KERRIGAN CT.
500                              SANTEE, CA  92071
READY.
```

New Deal

Dear Editor:

I am writing because I am flabbergasted by the inappropriateness of the solution given to the puzzle "It's in the Cards" in the October issue of *Creative Computing*.

Note that I did not say the solution is incorrect—it's just that a magazine devoted to computing ought to have presented a solution which is more natural to its environment.

My point is this: The puzzle is a natural representation of integers in binary notation. One can look at the cards as being constructed as follows. Card I contains all integers whose binary representation contains a 1 in the units (rightmost) position and is headed by number 1. Card II contains all integers whose 2's position (2nd rightmost) contains a 1 and is headed by number 2, etc.

Schematically, we may look at this as in the following example for the number 113 whose binary representation is 1110001.

VII	VI	V	IV	III	II	I
1	1	1	0	0	0	1

Hence the number appears on cards I, V, VI, VII leading to the equation $113 = 64 + 32 + 16 + 1$.

Incidentally, isn't the fact that any positive integer can be represented uniquely as a sum of distinct powers of 2, what the binary representation is all about?

N.S. Mendelsohn, Head
Department of Mathematics and Astronomy
The University of Manitoba
Winnipeg, Canada R3T 2N2

Bally Who?

Dear Editor:

I know of two active Bally home computer clubs. They help to disseminate programs written for Bally's limited storage memory, and both plan to overcome this shortfall by custom made external add-on units to be offered to the members in the near future.

I personally think the Bally unit has great potential and hope that Bally Mfg. will offer the add-on memory with Z-Grass graphics in Rom or software.

The Bally computer clubs are:

Arcadian
c/o Robert Fabris
3626 Morrie Dr.
San Jose, CA 95127

Cursor For Bally
P.O. Box 266
No. Hollywood, CA 91603

David R. Smikle
4553 Pinedale
Drayton Plains, MI 48020

Around and Around

Dear Editor:

In an article in *Creative Computing* ("Infinite Loop Finders Revisited," September 1979) I asked the question: "Is there anywhere, out there in the world, a *practical* infinite loop finder? If so, let me hear from you." Sure enough, Jan Hajek, who is at the Computer Center at Eindhoven (the Dutch technological institute which is the home base of Edsger Dijkstra, of structured programming fame) has written to me that APPROVER, a program of his, finds many kinds of infinite loops. Interested readers might write to him at THE-RC, P.O. Box 513, Eindhoven, The Netherlands.

W.D. Maurer
Professor
George Washington University
S.E.A.S.
Washington, D.C. 20052

Point/Counterpoint

Dear Editor:

I was excited to see the article in the August '80 issue about pie graphs. I have been working out some graphing programs for the Apple lately. The one problem that the author said he had was that when he plotted some colors next to each other (He gave the example of orange and green) he got strange results, and his dealer did not explain the cause of this in full detail.

The reason is as follows: each byte in hi-res represents 7 dots in black and white, or 4 dots in color. They are arranged like this (From MSB to LSB)

C d d d d d d d ,

where each "d" is one of the points on the screen, and the "C" is the color group mask. If it is a zero then the dots in that byte can only be black (0), white (3), green (1), or purple (2). If this bit is a one then the dots can only be black (4), white (7), orange (5), or blue (6)! By plotting a dot in the higher color group on a byte which is presently in the lower group or vice-versa), any previous dots were changed to a different color. I believe this is what the dealer refered to as color flip, and as far as I know there is no way to get around the problem.

I hope this cleared up any problems that Apple users have with this oddity. Keep up the good work.

Mark Kriegsman
103 Crest Dr.
Summit, NJ 07901

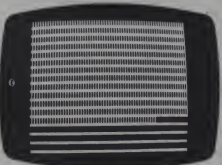
Fast Poke

Dear Editor:

Here is a dandy little graphics demo for the Level II TRS-80:

```
10 CLS: CLEAR
20 FOR A=129 TO 255
30 FOR X=15360 to 16383
40 IF INKEY$= "S" THEN 200
50 IF INKEY$= "C" THEN CLS
60 POKE X,A
70 NEXT X
80 X=0:NEXT A
90 CLS:INPUT"AGAIN (1/2)";AG
100 IF AG=1 THEN 10 ELSE END
110 REMARKABLE,HUH?
200 IF INKEY$= "R" THEN 50 ELSE 200
```

By poking into memory locations 15360 through 16383 we access the screen memory (each location in the screen memory represents a position on the videio monitor). We poke the TRS-80 CHR$ codes for graphics. The result is that the entire screen becomes fiilled with a pattern of graphics corresponding to the CHR$ code, then starts again from the top with a new pattern.



A few extra include:

1. Pressing "S" will hait program execution if you enjoy the current pattern.
2. Hitting "R" will start up where you left off before you hit "S".
3. Hitting "C" will clear the display and then continue on with the main program.

Jimmy Berkley
16 Fordham Road
Livingston, NJ 07039

Language Barrier

Dear Editor:

I have been a faithful subscriber to *Creative Computing* for some time now and have found a number of useful tips & programs for my TRS-80. However, I also use a Nova-3 at work and it can only run Fortran programs, which I cannot remember ever seeing in your magazine. Is there someplace where I can get some good Fortran programs for the Nova computer? (This is ANSII standard Fortran).

J.R. Fitzsimmons
Box 521 Star Rt.
Earleton, FL 32631

From time to time Creative has run programs in Fortran, for example "On Solving Alphametrics" by John Beidler, "Othello" by Ed Wright and several others. However, on surveys, reader response to Fortran pieces has been less-than-overwhelming to say the least. Hence, we've been running very little lately.
—DHA

No.18

Software with full support

Purchasing our software is just the beginning. We then back it up with professional support:

■ Subscription to "LIFELINES" for automatic notifications of revisions! ■ Update service for software and documentation! ■ Telephone Hotline! ■ Overseas software export service!

All Lifeboat programs require CP/M, unless otherwise stated.

Software with Manual / Manual Alone

CP/M® FLOPPY DISK OPERATING SYSTEM—Digital Research's operating system configured for many popular micro-computers and disk systems

System	Version	Price
Apple II® SoftCard® with Z80 Microsoft BASIC version 5 with high resolution graphics	2.x	349/NA
North Star Single Density	2.x	170/25 ®
North Star Double/Quad	2.x	170/25 ®
Durango F-85	2.x	170/25
iCOM Micro-Disk 2411	1.4	145/25
iCOM 3712 for MITS 88-2SIO Console	1.4	170/25 *
iCOM 3712 for 3P + S/MITS SIO Rev non-zero console	1.4	170/25 *
iCOM 3812	2.x	225/25 ®
iCOM 3812	1.4	170/25 *
iCOM 4511/Pertec D3000	2.x	375/25 *+
Mits 3202/Altair 8800	1.4	145/25
Heath H8 + H17	1.4	145/25 ®
Heath H89 by Magnolia	2.x	249/25
Ohio Scientific C3	2.x	200/25
Ohio Scientific C3-C	2.x	250/25
Onyx C8001 Standard	2.x	250/25
Onyx C8001 Enhanced	2.x	330/25
TRS-80 Model I	1.4	145/25 ®
TRS-80 Model II	2.x	170/25
TRS-80 Model II + Corvus	2.x	250/25
Processor Technology Helios II	1.4	145/25
Intel MDS Single Density	2.x	170/25
Intel MDS Double Density	2.x	170/25
New Micropolis Mod I	2.x	200/25 ®
New Micropolis Mod II	2.x	200/25 ®
Mostek MDX STD Bus System	2.x	350/25 **

HARD DISK OPTIONS (items with ®) when purchased with CP/M 2.X system

Corvus	Add 80.
Konan Phoenix	Add 80.
iCOM 4511/Pertec D3000	Add 80.

*Software consists of the operating system, text editor, assembler, debugger and other utilities for file management and system maintenance. Complete set of Digital Research's documentation and additional implementation notes included. Systems marked * and ** include firmware on 2708 and 2716. Systems marked + include 5440 media charge. Systems marked ® require the special ® versions of software in this catalog. + includes hardware addition to allow our standard versions of software to run under it.*

Z80 DEVELOPMENT PACKAGE—Consists of (1) disk file line editor, with global inter and intra-line facilities, (2) Z80 relocating assembler, Zilog/Mostek mnemonics, conditional assembly and cross reference table capabilities, (3) linking loader producing absolute Intel hex disk file **$95/$20**

ZDT—Z80 Monitor Debugger to break and examine registers with standard Zilog/Mostek mnemonic disassembly displays $35 when ordered with Z80 Development Package **$50/$10**

AVOCET SYSTEMS

XASM-68—Non-macro cross-assembler with nested conditionals and full range of pseudo operations. Assembles from standard Motorola MC6800 mnemonics to Intel hex **$200/$25**

XASM-65—As XASM-68 for MOS Technology MCS-6500 series mnemonics **$200/$25**

XASM-48—As XASM-68 for Intel MCS-48 and UPI-41 families **$200/$25**

XASM-18—As XASM-68 for RCA 1802 **$200/$25**

DISTEL—Disk based disassembler to Intel 8080 or TDL/Xitan Z80 source code listing and cross reference files. Intel or TDL/Xitan pseudo ops optional. Runs on 8080 **$65/$10**

DISILOG—As DISTEL to Zilog/Mostek mnemonic files **$65/$10**

SMAL/80 Structured Macro Assembler Language—Package of powerful general purpose text macro processor and SMAL structured language compiler. SMAL is an assembler language with IF-THEN-ELSE, LOOP-REPEAT-WHILE, DO-END, BEGIN-END constructs. Not compatible with CP/M version 2 or greater **$75/$15**

New **RAID**—8080/8085 debugging aid and general system utility program. Both a software emulator and real-time debug monitor. Features multiple breakpoints, symbolic input, symbolic display and altering of registers, built-in assembler & dis-assembler, tracing, single-stepping, memory protection, histograms, memory search, direct disk access. Requires 32K CP/M **$250/$25**

PHOENIX SOFTWARE ASSOCIATES

PASM®—Z80 macro assembler. Intel/TDL mnemonics. Generates Intel hex format or relocatable code in either TDL Object Module format or PSA Relocatable Binary Module format. Supports text insertion, conditional branching within macros, recursive macro calls and parameter passing **$129/$25**

EDIT—Character oriented text file editor. Includes macro definition capabilities. Handles insertion, deletion, searching, block move, etc. for files of any length. Does not require a CRT **$129/$25**

PLINK®—Two pass **disk-to-disk linkage editor/loader** which can produce re-entrant, ROMable code. Can link programs that are larger than available memory for execution targeted on another machine. Full library capabilities. Input can be PSA Relocatable Binary Module, TDL Object Module or **Microsoft REL files**. Output can be a COM file, Intel hex file, TDL Object Module or PSA Relocatable file **$129/$25**

BUG® and μBUG®—Z80 interactive machine level debugging tools for program development. BUG has full mnemonic trace and interactive assembly (mnemonics compatible with PASM). Dynamic breakpoints and conditional traps while tracing (even through ROM!). μBUG is a subset of BUG and is used in memory limited situations **$129/$25**

DIGITAL RESEARCH

MP/M—Installed for single density MDS-800. Multi-processing derivative of the CP/M operating system. Manual includes CP/M2 documentation **$300/$50**

MAC—8080 Macro assembler. Full Intel macro definitions. Pseudo Ops include RPC, IRP, REPT, TITLE, PAGE and MACLIB. Produces absolute hex output plus symbol table file for use by SID and ZSID (see below) **$120/$15**

SID—8080 Symbolic debugger. Full trace, pass count and breakpoint program testing. Has backtrace and histogram utilities. When used with MAC provides full symbolic display of memory labels and equated values **$105/$15**

ZSID—Z80 Symbolic debugger with all features of SID **$130/$15**

TEX—Text output formatter to create paginated, page-numbered and justified copy. Output can be directed to printer or disk **$105/$15**

DESPOOL—Utility program to permit simultaneous printing from text files while executing other programs **$80/$10**

tiny C—Interactive interpretive system for teaching structured programming techniques. Manual includes full source listings **$105/$50**

BDS C COMPILER—Supports structures, unions, 2 dimensional arrays, pointers, recursion and overlays. Features optimized code generator, variable sized buffers for file I/O and capability to produce ROMable code. Includes macro package to enable user to produce linkable modules with MAC (see under Digital Research). Floating point functions, full run-time package and machine code library sources provided. Linker, library manager and textbook included. Compiler lacks initializers, statics, floats and longs **$145/$25**

WHITESMITHS C COMPILER—The ultimate in systems software tools. Produces faster code than a pseudo-code Pascal with more extensive facilities. Conforms to the full UNIX® Version 7 C language, described by Kernighan and Ritchie, and makes available over 75 functions for performing I/O, string manipulation and storage allocation. Linkable to Microsoft REL files. Requires 60K CP/M **$630/$30**

MICROSOFT

Software with Manual / Manual Alone

BASIC-80—Disk Extended BASIC, ANSI compatible with long variable names, WHILE/WEND, chaining, variable length file records. MBASIC version 4.51 also included on disk **$325/$25**

BASIC COMPILER—Language compatible with BASIC-80 and 3-10 times faster execution. Produces standard Microsoft relocatable binary output. Includes MACRO-80. Also linkable to FORTRAN-80 or COBOL-80 code modules **$350/$25**

FORTRAN-80—ANSI 66 (except for COMPLEX) plus many extensions. Includes relocatable object compiler, linking loader, library with manager. Also includes MACRO-80 (see below) **$425/$25**

COBOL-80—Level 1 ANSI 74 standard plus most of Level 2. Full sequential, relative, and indexed file support with variable file names. Powerful interactive, formatted screen handling with ACCEPT and DISPLAY verbs. Program segmentation for execution of programs larger than memory and CHAIN command with parameter passing. Full support of CP/M version 2 files. Includes MACRO-80 (see above), linking loader and relocatable library manager. Requires 48K CP/M **$700/$25**

New **M/SORT**—Optional sort/merge capability for COBOL-80 which conforms fully to SORT/MERGE, Level II of the 1974 ANSI COBOL standard (except COLLATING SEQUENCE IS alphabet-name). Requires COBOL-80. Sold as an update to COBOL-80 **$150/$10**
COBOL-80 + M/SORT **$825/$35**

MACRO-80—8080/Z80 Macro Assembler. Intel and Zilog mnemonics supported. Relocatable linkable output. Loader, Library Manager and Cross Reference List utilities included **$149/$15**

XMACRO-86—8086 cross assembler. All Macro and utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available **$275/$25**

EDIT-80—Very fast random access text editor for text with or without line numbers. Global and intra-line commands supported. File compare utility included **$89/$15**

muSIMP/muMATH—muSIMP is a high level programming language suitable for symbolic and semi-numerical processing implemented using a fast and efficient interpreter requiring only 7K bytes of machine code. muMATH is a package of programs written in muSIMP. The package performs sophisticated mathematical functions. Keeps track of up to 611 digits. Performs matrix operations on arrays: transpose, multiply, divide, inverse and other integer powers. Logarithmic, exponential, trigonometric simplification and transformation, symbolic differentiation with partial derivatives, symbolic integration of definite and indefinite integrals. Requires 40K CP/M **$250/$20**

muLISP-80—Microcomputer implementation of LISP. The interpreter resides in only 7K bytes of memory yet includes 83 LISP functions. Has infinite precision integer arithmetic expressed in any radix from 2 to 36. muLISP80 includes complete trace facility and a library of useful functions and entertaining sample programs **$200/$15**

PASCAL/M®—Compiles enhanced Standard Pascal to compressed efficient Pcode. Totally CP/M compatible. Random access files. Both 16 and 32-bit integers. Runtime error recovery. Convenient STRINGs. OTHERWISE clause on CASE. Comprehensive manual (90 pp. indexed). SEGMENT provides overlay structure. INPORT, OUTPORT and untyped files for arbitrary I/O. Requires 56K CP/M. Specify 1) 8080 CP/M, 2) Z80 CP/M, or 3) Cromamco CDOS **$175/$20**

PASCAL/Z—Z80 native code PASCAL compiler. Produces optimized, ROMable re-entrant code. All interfacing to CP/M is through the support library. The package includes compiler, relocating assembler and linker, and source for all library modules. Variant records, strings and direct I/O are supported. Requires 56K CP/M **$395/$25**

New **PASCAL/MT**—Subset of standard PASCAL. Generates ROMable 8080 machine code. Symbolic debugger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Includes strings, enumerations and record data types. Manual explains BASIC-PASCAL conversion. Requires 32K **$250/$30**

APL/V80—Concise and powerful language for application software development. Complex programming problems are reduced to simple expressions in APL. Features include up to 27K active workspace, shared variables, arrays of up to 8 dimensions, disk workspace and copy object library. The system also supports auxiliary processors for interfacing I/O ports. Requires 48K CP/M and serial APL printing terminal or CRT **$500/$30**

ALGOL-80—Powerful block-structured language compiler featuring economical run-time dynamic allocation of memory. Very compact (24K total RAM) system implementing almost all Algol 60 report features plus many powerful extensions including string handling, direct disk address I/O etc **$199/$20**

CBASIC-2 Disk Extended BASIC—Non-interactive BASIC with pseudo-code compiler and run-time interpreter. Supports full file control, chaining, integer and extended precision variables etc. Versions of CRUN for CP/M versions 1.4 and 2.x included on disk **$120/$15**

MICRO FOCUS

Software with Manual / Manual Alone

STANDARD CIS COBOL—ANSI 74 COBOL standard compiler fully validated by U.S. Navy tests to ANSI level 1. Supports many features to level 2 including dynamic loading of COBOL modules and a full ISAM file facility. Also, program segmentation, interactive debug and powerful interactive extensions to support protected and unprotected CRT screen formatting from COBOL programs used with any dumb terminal **$850/$50**

FORMS 2—CRT screen editor. Output is COBOL data descriptions for copying into CIS COBOL programs. Automatically creates a query and update program of indexed files using CRT protected and unprotected screen formats. No programming experience needed. Output program directly compiled by STANDARD CIS COBOL **$200/$20**

New **NEVADA COBOL**—Subset of ANSI-74. Features fast compilation and execution with small object modules. Has extended arithmetic with 18 digit accuracy. Extended I/O includes random access files and sequential files of both fixed and variable length records, and interactive accept/display verbs. Good error messages and debugging facilities enhance program development. Requires a 32K CP/M system **$149/$25**

EIDOS SYSTEMS

KBASIC—Microsoft Disk Extended BASIC version 4.51 integrated with KISS Multi-Keyed Index Sequential and Direct Access file management as 9 additional BASIC commands. KISS included as relocatable modules linkable to FORTRAN-80, COBOL-80, and BASIC COMPILER. Specify CP/M version 1.4 or 2.x when ordering. Requires 48K CP/M **$585/$45**
To licensed users of Microsoft BASIC 80 (MBASIC) **$435/$45**

XYBASIC Interactive Process Control BASIC—Full disk BASIC features plus unique commands to handle byte rotate and shift and to test and set bits. Available in several versions

Integer ROM squared	$350/$25
Integer CP/M	$350/$25
Extended ROM squared	$450/$25
Extended CP/M	$450/$25
Extended Disk CP/M	$550/$25
Integer CP/M Run Time Compiler	$350/$25
Extended CP/M Run Time Compiler	$450/$25

RECLAIM—A utility to validate media under CP/M. Program tests a diskette or hard disk surface for errors, reserving the imperfections in invisible files, and permitting continued usage of the remainder. Essential for any hard disk. Requires CP/M version 2 **$80/$5**

BASIC UTILITY DISK—Consists of: (1) CRUNCH-14—Compacting utility to reduce the size and increase the speed of programs in Microsoft BASIC 4.51, BASIC-80 and TRS-80 BASIC. (2) DPFUN—Double precision subroutines for computing nineteen transcendental functions including square root, natural log, log base 10, sine, arc sine, hyperbolic sine, hyperbolic arc sine, etc. Furnished in source on diskette and documentation **$50/$35**

STRING/80—Character string handling plus routines for direct CP/M BDOS calls from FORTRAN and other compatible Microsoft languages. The utility library contains routines that enable programs to chain to a COM file, retrieve command line parameters and search file directories with full wild card facilities. Supplied as linkable modules in Microsoft format **$95/$20**

STRING/80 source code available separately— **$295/NA**

THE STRING BIT—FORTRAN character string handling. Routines to find, fill, pack, move, separate, concatenate and compare character strings. This package completely eliminates the problems associated with character string handling in FORTRAN. Supplied with source **$65/$15**

VSORT—Versatile sort/merge system for fixed length records with fixed or variable length fields. VSORT can be used as a stand-alone package or loaded and called as a subroutine from CBASIC-2. When used as a subroutine, VSORT maximizes the use of buffer space by saving the TPA on disk and restoring it on completion of sorting. Records may be up to 255 bytes long with a maximum of 5 fields. Upper/lower case translation and numeric fields supported **$175/$20**

New **IBM/CPM**—Program to transfer IBM 3741 data set files to CP/M files or CP/M files to IBM 3741 data sets. Easy to use. Requires two eight inch diskette drives, 24K memory, and a 24 by 80 CRT terminal **$175/$5**

CPAids®

MASTER TAX—Professional tax preparation program. Prepares schedules A, B, C, D, E, F, G, R/RP, SE, TC, ES and forms 2106, 2119, 2210, 3468, 3903, 2441, 4625, 4726, 4797, 4972, 5695 and 6251. Printing can be on readily available, pre-printed continuous forms, on overlays, or on computer generated, IRS approved forms. Maintains client history files and is interactive with CPAids GENERAL LEDGER II (see below) **$995/$30**
Annual Update Fee **$350**

STANDARD TAX—As above for schedules A, B, C, D, E, G, R/RP, SE, TC and forms 2106 and 2441. Also, does not maintain client history files **$495/$30**
Annual Update Fee **$175**

Neu in der Schweiz Lifeboat Associates GmbH, Aegeristr. 35, CH 6340 Baar **Telefon 042/31 2931**

Software with Manual / Manual Alone

GENERAL LEDGER II — Designed for CPAs. † Stores complete 12 month detailed history of transactions. Generates financial statements, depreciation, loan amortizations, journals, trial balances, statements of changes in financial position, and compilation letters. Includes payroll system with automatic posting to general ledger. Prints payroll register, W2's and payroll checks. **$450/$30**

T/MAKER — Powerful new tool for preparing management reports with tabular data. Makes financial modeling projects easy. Do you want a weekly profitability report? Set up the table and compute. Just change the sales figures for next week and compute. You have a new report! T/MAKER includes a full screen editor for setting up tables which pages left, right, up and down. Compute includes standard arithmetic, percents, exponents, common transcendental functions, averages, maxima, minima, projections, etc. Requires 48K CP/M, CBASIC-2, CRT terminal with addressable cursor positioning. **$275/$25**

New **ESQ-1** — Professional time and billing for the legal profession. Designed for use by the first-time computer user. Records billable and non-billable time. Complete system includes transaction entry, posting, billing, reports, and client analysis. Records cash receipts, escrow receipts, and escrow transfers. Requires 48K CP/M system, 480K of disk storage space, cursor addressable CRT, and CBASIC-2. **$1495/$50**

Complete demonstration system for ESQ-1 **$75/$50**

BSTAM — Utility to link one computer to another also equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansion to send *.COM, etc. 9600 baud with wire. 300 baud with phone connection. **Both** ends need one. Standard and ® versions can talk to one another. This software requires a knowledge of assembler language for installation. **$150/$10**

New **BSTMS** — Intelligent terminal program for CP/M systems. Permits communication between micros and mainframes. Sends character data files to remote computers under complete control. System can record character data sent from remote computer systems and data banks. Includes programs to EXPAND and COMPRESS binary files for transmission. This software requires a knowledge of assembler language for installation. **$200/$25**

WHATSIT?* — Interactive data-base system using associative tags to retrieve information by subject. Hashing and random access used for fast response. Requires CBASIC-2. **$175/$25**

SELECTOR III-C2 — Data Base Processor to create and maintain multi-key data bases. Prints formatted sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments, etc. Requires CBASIC-2. Supplied in source. **$295/$20**

GLECTOR — General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2, CBASIC-2 and 58K system. **$350/$25**

New **MAGSAM III** — Sophisticated keyed access file support system. Supports random, sequential, and generic retrieval by key. Also, multiple secondary keys. Dynamic allocation and extension of files with automatic free space reclamation. Interactive tutorial included to get the user started. Complete with documentation and quick reference card. Specify CBASIC or Microsoft BASIC version. Requires 48K system. **$145/$25**

New **MAGSAM IV** — High speed machine code version of MAGSAM III for CBASIC only. Distributed as pre-loaded modules and Microsoft relocatable object modules. **$295/$25**

DMA

CBS — Configurable Business System is a comprehensive set of programs for defining custom data files and application systems without using a programming language such as BASIC, FORTRAN, etc. Multiple key fields for each data file are supported. Set-up program customizes system to user's CRT and printer. Provides fast and easy interactive data entry and retrieval **with transaction processing**. Report generator program does complex calculations with stored and derived data, record selection with multiple criteria, and custom formats. Sample inventory and mailing list systems included. **No support language required.** **$395/$40**

MICROPRO

SUPER-SORT I — Sort, merge, extract utility as absolute executable program or linkable module in Microsoft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating & fixed point exponential, field justified, etc. Even variable number of fields per record! **$225/$25**

SUPER-SORT II — Above available as absolute program only. **$175/$25**

SUPER-SORT III — As II without SELECT/EXCLUDE. **$125/$25**

Software with Manual / Manual Alone

DATASTAR — Professional forms control entry and display system for key-to-disk data capture. Menu driven with built-in learning aids. Input field verification by length, mask, attribute (i.e. upper case, lower case, numeric, auto-dup, etc.). Built-in arithmetic capabilities using keyed data, constant and derived values. Visual feedback for ease of forms design. Files compatible with CP/M-MP/M supported languages. Requires 32K CP/M and CRT with addressable cursor. **$350/$35**

WORD-STAR — Menu driven visual word processing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace, Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning. **$445/$40**

WORD-STAR-MAIL-MERGE — As above with option for production mailing of personalized documents with mail lists from DATASTAR or NAD. **$575/$40**

WORD-MASTER Text Editor — In one mode has superset of CP/M's ED commands including global searching and replacing, forwards and backwards in file. In video mode, provides full screen editor for users with serial addressable-cursor terminal. **$145/$25**

MAGIC WAND* — Word processing system with simple, easy to use full screen text editor and powerful print processor. Editor has all standard editing functions including text insert and delete, global search and replace, block move and library files for boiler plate text. Print processor formatting commands include automatic margins, pagination, headings & footings, centered and justified text. Also prints with true proportional spacing, merges with data files for automatic form letters, and performs run-time conditional testing for varied output. Requires 32K CP/M and CRT terminal with addressable cursor. **$395/$40**

TEXTWRITER III — Text formatter to justify and paginate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil* and Word-Star prepared files. **$125/$20**

New **DATEBOOK** — Program to manage time just like an office appointment book but using the speed and memory of a computer. Keeps track of three appointment schedules (three dental chairs, three attorneys, etc.) at once. Appointments consist of name, reason for the appointment, the date and time, and the length of the appointment. System can be quickly customized for the individual user. Many helpful features for making, changing, finding, and reporting appointments. Requires 48K CP/M and 180K bytes diskette storage. Requires 80 × 24 cursor addressable terminal. Specify 8080 CP/M, Z80 CP/M or Cromemco CDOS. **$295/$25**

PEACHTREE SOFTWARE*

General accounting software for small businesses. Each product can be used alone or with automatic posting to the General Ledger. Supplied in source for Microsoft BASIC 4.51.

GENERAL LEDGER	$530/$40
ACCOUNTS PAYABLE	$530/$40
ACCOUNTS RECEIVABLE	$530/$40
PAYROLL	$530/$40
INVENTORY	$660/$40

Other application products supplied in source for Microsoft BASIC 4.51

MAILING ADDRESS	$530/$40
PROPERTY MANAGEMENT	$925/$40

GRAHAM-DORIAN SOFTWARE SYSTEMS

Comprehensive accounting software written in CBASIC-2 and supplied in source code. Each software package can be used as a stand-alone system or integrated with the General Ledger for automatic posting to ledger accounts. Requires CBASIC-2.

GENERAL LEDGER	$805/$40
ACCOUNTS PAYABLE	$805/$40
ACCOUNTS RECEIVABLE	$805/$40
INVENTORY SYSTEM	$555/$40
JOB COSTING	$805/$40
APARTMENT MANAGEMENT	$805/$40
CASH REGISTER	$805/$40

POSTMASTER — A comprehensive package for mail list maintenance that is completely menu driven. Features include keyed record extraction and label production. A form letter program is included which provides neat letters on single sheet or continuous forms. Includes NAD file translator. Requires CBASIC-2. **$150/$20**

STRUCTURED SYSTEMS GROUP

Complete interactive accounting software for business. Each product can be used stand-alone or with automatic posting to the general ledger. Each product is thoroughly tested and very well documented.

GENERAL LEDGER	$820/$40
ACCOUNTS RECEIVABLE	$820/$40
ACCOUNTS PAYABLE	$820/$40
PAYROLL	$820/$40
INVENTORY CONTROL	$820/$40

Software with Manual / Manual Alone

ANALYST — Customized data entry and reporting system. User specifies up to 75 data items per record. Interactive data entry, retrieval, and update facility makes information management easy. Sophisticated report generator provides customized reports using selected records with multiple level break-points for summarization. Requires a disk sort utility such as QSORT, SUPER-SORT or VSORT and CBASIC-2. **$250/$15**

LETTERIGHT — Program to create, edit and type letters or other documents. Has facilities to enter, display, delete and move text, with good video screen presentation. Integrates with NAD for form letter mailings. **$200/$25** New

NAD — Name and Address selection system. Interactive mail list creation and maintenance program with output as full reports with reference data or restricted information for mail labels. Transfer system for extraction and transfer of selected records to create new files. QSORT required if sorting is desired. **$100/$20**

QSORT — Fast sort/merge program for files with fixed record length, variable field length information. Up to five ascending or descending keys. Full back-up of input files created. **$100/$20**

★ ★ ★ ★ ★ ★ ★

HEAD CLEANING DISKETTE — Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use. Specify 5" or 8".

Single sided	$20 each/$55 for 3
Double sided	$25 each/$65 for 3

LIFELINES
NEWSLETTER FROM LIFEBOAT

LIFELINES is the first step in software support for the serious microcomputer user. Each issue reports new revisions together with information on the purpose for each such release, be it for correction of bugs or the addition of features and facilities.

Feature Articles | New Software | Product Comparisons | Info on CP/M Users Group |

SUBSCRIPTION INFORMATION:

$18 for twelve issues U.S., Canada, and Mexico
$40 for twelve issues, all other countries
$2.50 for each back issue U.S., Canada and Mexico
$3.80 for each back issue, all other countries

Send Check to LIFELINES, 1651 Third Avenue, New York, N.Y. 10028 or use your VISA or MASTERCARD — call (212) 722-1700

DC 300 Data Cartridges Specify 450 XL or 300 certified. Pack of 5 **$100**

FLIPPY DISK KIT — Template and instructions to modify single sided 5¼" diskettes for use of second side in single sided drives. **$12.50**

FLOPPY SAVER — Protection for center holes for 5" and 8" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool and tough 7 mil mylar reinforcing rings for 25 diskettes.

5", Kit	$14.95
5", Rings only	$7.95
8", Kit	$16.95
8", Rings only	$8.95

New **The CP/M HANDBOOK** (with MP/M) by Rodnay Zaks **$13.95**

PASCAL USER MANUAL AND REPORT — By Jensen and Wirth. The standard textbook on the language. Recommended for use by Pascal/Z, Pascal/M and Pascal/MT users. **$12**

THE C PROGRAMMING LANGUAGE — By Kernighan and Ritchie. The standard textbook on the language. Recommended for use by BDS C, tiny C, and Whitesmiths C users. **$12**

STRUCTURED MICROPROCESSOR PROGRAMMING — By the authors of SMAL/80. Covers structured programming, the 8080/8085 instruction set and the SMAL/80 language. **$20**

ACCOUNTS PAYABLE & ACCOUNTS RECEIVABLE — CBASIC book by Osborne/McGraw-Hill **$20**

GENERAL LEDGER — CBASIC book by Osborne/McGraw-Hill **$20**

PAYROLL WITH COST ACCOUNTING — CBASIC book by Osborne/McGraw-Hill **$20**

★ ★ ★ ★ ★ ★ ★ ★

* Program names trademarked

† Recommended system configuration consists of 48K CP/M, 2 full size disk drives, 24 x 80 CRT and 132 column printer

® Modified version available for use with CP/M as implemented on Heath and TRS-80 Model I computers

① User license agreement for this product must be signed and returned to Lifeboat Associates before shipment may be made

① This product Includes/eXcludes the language manual recommended in Condiments

⑤ Serial number of CP/M system must be supplied with orders

② Requires Z80 CPU

★ ★ ★ ★ ★ ★ ★ ★

Ordering Information

MEDIA FORMAT ORDERING CODES.
When ordering, please specify format code.

LIFEBOAT ASSOCIATES MEDIA FORMATS LIST. Diskette, cartridge disk and cartridge tape format codes to be specified when ordering software for listed computer or disk systems. All software products have specific requirements in terms of hardware or software support, such as MPU type, memory size, support operating system or language.

Computer system	Format Code
Altair 8800 Disk	See MITS 3200
Altos	A1*
Apple · SoftCard 13 Sector	RD
Apple · SoftCard 16 Sector	RR
AVL Eagle	RB
BASF System 7100	RD
Blackhawk Single Density	Q3
Blackhawk Micropolis Mod II	Q2
CDS Versatile 3B	Q1
CDS Versatile 4	Q2
COMPAL-80	Q2
Cromemco System 3	A1*
Cromemco Z2D	R6
CSSN BACKUP (tape)	T1#
Delta	A1*
Digi-Log Microterm II	RD
Digital Microsystems	A1*
Discus	See Morrow Discus
Durango F-85	RL
Dynabyte DB8/2	R1
Dynabyte DB8/4	A1*
Exidy Sorcerer · Lifeboat CP/M	Q2
Exidy Sorcerer · Exidy CP/M	Q4
Heath H8 · H17/H27	P4
Heath H89 · Lifeboat CP/M	P4
Heath H89 · Magnolia CP/M	P7
Helios II	See Processor Technology
Horizon	See North Star
iCOM 2411 Micro Floppy	R3
iCOM 3712	A1
iCOM 3812	A1*
iCOM 4511 5440 Cartridge CP/M 1.4	D1#
iCOM 4511 5440 Cartridge CP/M 2.2	D2#
IMS 5000	RA
IMS 8000	A1*
IMSAI VDP-40	R4**
IMSAI VDP-42	R4**
IMSAI VDP-44	R5**
IMSAI VDP-80	A1**
Intecolor	See ISC Intecolor
Intel MDS Single Density	A2
Intel MDS Double Density	A5
Intertec SuperBrain DOS 0.1	R7
Intertec SuperBrain DOS 0.5-2.X	RJ
Intertec SuperBrain DOS 3.X	RK
ISC Intecolor 8063/8360/8963	A1
Kontron PSI-80	RF
Meca 5¼	P6
Micromation (Except TRS-80 below)	A1*
Micropolis Mod I	Q1
Micropolis Mod II	Q2
MITS 3200/3202	B1
Morrow Discus	A1*
Mostek	A1
MSD 5¼	RC
North Star Single Density	P1
North Star Double/Quad	P2
Nylac Single Density	Q3
Nylac Micropolis Mod II	Q2
Ohio Scientific C3	A3
Onyx C8001	T2#
Pertec PCC 2000	A1*
Processor Technology Helios II	B2
Quay 500	RO
Quay 520	RP
RAIR Single Density	R9
RAIR Double Density	RE
Research Machines 8	A1
Research Machines 5¼*	RH
REX	Q3
Sanco 7000 5¼	RQ
SD Systems 8	A1*
SD Systems 5¼	R3
Sorcerer	See Exidy Sorcerer
Spacebyte	A1
SuperBrain	See Intertec
Tarbell	A1*
TEI 5¼	R3
TEI 8	A1*
Thinkertoys	See Morrow Discus
TRS-80 Model I 5¼	R2
TRS-80 Model I · FEC Freedom	RN
TRS-80 Model I · Micromation	A4*
TRS-80 Model I · Omikron 5¼	RM
TRS-80 Model I · Omikron 8*	A1
TRS-80 Model I · Shuffleboard 8	A1
TRS-80 Model II	A1*
VDP-40/42/44/80	See IMSAI
Vector Graphic	Q2
Vector MZ	Q2
Versatile	See CDS Versatile
Vista V80 5¼ Single Density	P5
Vista V200 5¼ Double Density	P6
Zenith Z89 · Lifeboat CP/M	P4
Zenith Z89 · Magnolia CP/M	P7

Prices reflect distribution on 8" single density diskettes. If a format is requested which requires additional diskettes, a surcharge of $8 per additional diskette will be added.

Prices F.O.B. New York. Shipping, handling and C.O.D. charges extra.

Manual cost applicable against price of subsequent software purchase.

The sale of each proprietary software package conveys a license for use on one system only.

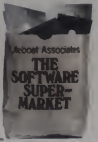
VISA · MasterCharge

* Single-Side Single-Density disks are supplied for use with Double-Density and Double Side 8" soft sector format systems.

** IMSAI formats are single density with directory offset of zero.

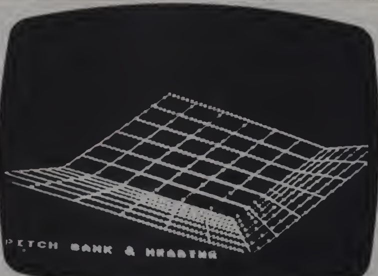
\# A media surcharge of $25 for orders on tape formats T1 and T2 and of $100 for orders on disk formats D1 and D2 will be added.

The list of available formats is subject to change without notice. In case of uncertainty call to confirm the format code for any particular equipment.



Lifeboat Associates, 1651 Third Avenue, N.Y., N.Y. 10028 **(212) 860-0300**
International Telex: 220501, Domestic Telex: 640693

™The Software Supermarket is a trademark of Lifeboat Associates



Soft Centered

David Lubar

Barring plague, famine, or a chromium-oxide shortage, this will be a regular feature, acting as a catch-all for reviews of software, bits of news, and whatever else seems worth passing on. This month, a tough game, a good utility, and a taxing program.

50 Ways to Leave Your Life

Med Systems Software, P.O. Box 2674, Chapel Hill NC 27541, has come out with *Deathmaze 5000*, a tape for a 16K TRS-80 ($12.95). They are also working on a 32K Apple version that will run on either Apple. The game is an adventure that takes places in a large maze of corridors with five levels. The display consists of nicely-done three-dimensional graphics of the sort used in *Tunnel Vision*. Since the program is written in machine language, the screen changes almost instantly as you walk through the corridors. The player can either move through the maze using arrow keys, or give two-word commands such as "Take knife." Throughout the corridors, there are boxes containing objects. The purpose of the game is to get out alive. This is not an easy task; there are many ways to get beheaded, smashed, crushed, or otherwise removed from play. The game can be saved on tape. Unfortunately, attempts to reload the data left me in other areas of the maze. Despite this, the game is very good, though I still haven't figured out how to progress beyond the first level. Med Systems had graciously provided all the instructions necessary to let you put the game on disk. They haven't provided instructions for how to get past the first level. Such is life.

A New Perspective

An excellent utility for the Atari is *3-Dimensional Computer Graphics* ($29.95), which will run on either the 400 or 800. From **Sebree's Computing**, 456 Granite Ave., Monrovia CA 91016, the set of four programs allows the drawing of wire-frame type objects. The objects can be viewed at different angles, moved along any of the coordinates or rotated. The programs run with as little as 8K, but higher resolutions require more memory. The first program demonstrates the abilities of the system, allowing the user to input x,y, and z cordinated as well as pitch, bank, and heading. Then a prestored figure is drawn. The second program allows you to define figures. The last two programs demonstrate ways to incorporate the drawings into full scenes. Whether vou just want to play around with three-dimensional drawing, or need a graphics utility for working with solid objects, this package is worth considering.

Giving Sam His Dues

Micromatic Programming Company, 24 Old Farm Road, Weston CT 06829, sent me a demonstration sample of their *Tax|Saver* program, which will be released this January as a set of disks or cassettes for the TRS-80. While the sample didn't contain the full series of programs, it had enough to show that Tax/Saver is well designed and easy to use. Basically, the user is asked a series of questions, allowing the program to determine such things as filing status, required forms, etc. You have to have all the necessary information on hand. The program has some nice features. For instance, it will inform you if your employer is deducting too much for Social Security. If you are filing a joint return, it will ask if you also want to try doing forms for filing separately, thus allowing you to determine which way costs less. At various points, tricky questions are explained, either in the manual or in the program. For example, if you want to declare dependents, there is a program that helps you determine whether each individual in question can qualify as a dependent. Once all data has been entered, the program displays each line of the appropriate form and shows what should be entered in that line.

Obviously, not everyone needs this type of help. If your return is very simple, there isn't much for the computer to do. And if your return is very complicated, you probably have an accountant. But if you are somewhere in the middle, and the thought of filling out those forms sends shivers through your body, this program could ease the painful process of paying Uncle Sam.

Scattered Bits

It seems that the shortage of Atari software is coming to an end. Many companies are moving into this area, both with conversions of existing programs, and with original material. At a slower rate, the same thing is happening for the TI 99/4. TRS-80 owners will be pleased to learn that **Adventure International** is working on a new series of games which combine Adventure with Dungeons and Dragons. The first game of the series will require dual disks, further releases will run on smaller configurations. These will be reviewed as soon as they are available. The first American software for the Sinclair ZX 80 should be hitting the market soon, courtesy of **Image**. Several new releases from **Creative Computing** will be available in the near future, including an excellent version of *Milestones* for the Apple.

Parting Words

The problem of software piracy affects everyone who is involved with computers. A related issue is protected software. How do you, the users, feel about uncopyable programs? Are there alternatives? Would you rather pay more and receive a copyable program? Your thoughts are welcome. □

NOW THE SOFTCARD CAN TAKE YOU BEYOND THE BASICS.

[illegible] Z-80® [illegible] CP/M®.

You may even know that with the SoftCard, you get Microsoft's powerful BASIC — extended to support Apple graphics and many other features.

Now, whenever you're ready to get beyond the BASICs, the SoftCard can take you into whole new realms. Starting with two advanced language packages from Microsoft.

FORTRAN AND COBOL TO GO.

Now you can run the world's most popular engineering/scientific language and the most popular business language on your Apple. Think what that means: you can choose from literally thousands of "off-the-shelf" applications programs, and have them working with little conversion. Or design your own programs, taking advantage of all the problem-solving power these specialized languages give you.

FORTRAN-80

A complete ANSI-standard FORTRAN (except COMPLEX type), with important enhancements. The extremely fast compiler performs extensive code [illegible] Apple FORTRAN.

FORTRAN is easy to learn if you know BASIC and the package includes a huge library of floating point, math, and I/O routines you can use in all your programs.

COBOL-80

Virtually the only choice for serious business data processing. It's ANSI 1974 standard COBOL, with many user-oriented features added: formatted screen support for CRT terminals, simple segmenting of very large programs, powerful file handling capability, trace debugging, and much more. A separate Sort package is coming soon.

FORTRAN-80 and COBOL-80 are just two more reasons why the Apple with SoftCard is the world's most versatile personal computer. Get all the exciting details from your Microsoft dealer today. And start getting beyond the BASICs.

MICROSOFT Consumer Products, 400 108th Ave. N.E., Suite 200, Bellevue, WA 98004. (206) 454-1315.

MICROSOFT

CIRCLE 168 ON READER SERVICE CARD

TRS-80 COLOR COMPUTER

James Garon

Documentation

Three pieces of documentation come with the computer: A 151 page beginner's book called *Getting Started With Color Basic*, A 31 page *Color Computer Operation Manual*, and a Quick Reference Card.

Getting Started With Color Basic is an excellent introduction to Color Basic. While it does not attempt to cover all aspects of programming, or even the complete instruction set of Color Basic, it does take a core vocabulary and explain it well. The style is similar to Learning Level I; there are abundant humorous illustrations, clear examples, and well-thought-out diagrams which manage to convey both simple and complex concepts in a manner which will not intimidate the newcomer to computing. After a concept has been presented, the reader is invited to write a "Do it yourself program." Sample solutions are given in the text for the shorter exercises and an entire appendix is devoted to solutions to the longer programs. It is interesting to note that while semicolons are often optional in both Level II and Color Basic PRINT statements, this is the first manual that tacitly acknowledges the fact.

The *Color Computer Operation Manual* covers topics such as connecting the computer to your television, connection and use of joysticks, printer, program cartridges, and cassette recorder. Here also, the programmer already familiar with Basic will find information on the special features of Color Basic, such as producing sounds and creating graphics.

The Quick Reference Card (8½ x15", printed on both sides) is divided into eight sections:

•Start-UP: Tells you how to turn the system on.
•Special characters (' $: ? , ;) all have the same meanings as in Level II.
•Operators (+,—,*, , , AND, OR etc. indicating the order in which they are evaluated by Basic).
•Basic Functions: There are 22 functions listed alphabetically, from ABS to VAL.)
•Control Keys: There are eight, seven of which the Level II programmer will be familiar with; left-arrow to erase one character, shift-left-arrow to erase a line being typed, BREAK, CLEAR, ENTER, the spacebar, and SHIFT-@ to halt a listing or program execution until another key has been pressed (unlike Level II however, you must use a key other than SHIFT-@ to resume a listing or halted program). The eighth control key is SHIFT-0 which is used to toggle back and forth between upper and lowercase (lowercase is shown as reverse video uppercase: black letters on a green background. These reversed characters will appear as lower case on a lineprinter).
•Basic Statements: 41 statements, again listed alphabetically, from AUDIO to STOP.
•Video Control Codes (there are only three: decimal 8, 13 and 32 which perform similarly to their Level II cousins: backspace and erase, linefeed and carriage return, and a blank.
•Error Messages: there are 25 2-character codes.

In addition to the above, there are only three more items in the box: a 12 foot cable for connecting the computer to your television set, a switch box that allows you to select either the computer signal or television programs, and the computer itself. The computer is attractively styled in silver and black with simulated mother-of-pearl and colored accents.

The only cause for complaint might be the keys themselves, which look and sound much like rectangular pieces of hard candy or gum: however, after a few minutes of typing one gets used to their feel and their clatter. The keys are color-coded; most keys are gray, the control keys are white, and the powerful BREAK key is red.

In the rear of the cabinet are five jacks, two buttons, a switch, and a power cord. One jack accepts the cable which carries sound and picture to your television. The next jack is for the optional CTR-80A recorder cable. (The cable from a regular TRS-80 recorder will not fit here.) The third jack is for an RS-232-C device such as a serial printer or possibly a modem. The last two jacks accept one joystick each.

The two buttons are located at either

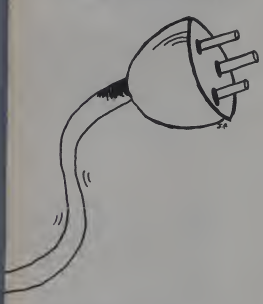
James Garon, 46 Kinsley St., Nashua, NH 03060.

side of the back panel; one is the power switch, the other is the RESET button. Both are safe from accidental pressing, yet easily accessible without having to be a finger-contortionist as is necessary with the original TRS-80. The switch allows the selection of television channel 3 or channel 4, whichever is weakest in your area.

The power cord is just that—a power cord; there is no awkward power supply enclosure to clutter your work area as there is with the original TRS-80. It has a three prong, grounded plug which caused a few brief moments of panic since our eight year old New Hampshire apartment, while it has many outlets, has only one three-holer and the original TRS-80 is plugged into that! Borrowing the wife's ironing extension cord and leaving the ground plug hanging helplessly out in space solved the problem long enough for this review to be completed. (Good thing there were no lightning storms this week!)

Let's look at some negative features first, and save the good stuff for dessert:

The display format is very small; the



screen can display only 512 characters in 16 rows of 32.

There is no way to edit a line once it has been entered other than by retyping it. While a line can be as long as 249 characters, most programmers will not want to pack a line this tightly since a single typing error will require that the entire line be retyped.

Variables on either side of the word "TO" in a FOR/NEXT loop must be separated from that word by at least one space:

```
10 FOR I = ATOB
```

will not work properly. Instead, use:

```
10 FOR I = A TO B
```

Those who are used to the compressed form (without spaces) allowed in Level II will find this difficult to remember at first, Color Basic does not recognize the word "LET." This means that you must replace lines such as:

```
10 LET A=7
```

with:

```
10 A=7.
```

Since most programmers do not use the word "LET" anyway, this is not a particular hardship. The word "THEN" may not be omitted (or replaced by a comma). A line such as:

```
10 IF X=Y PRINT "YES"
```

will need to be rewritten:

```
10 IF X=Y THEN PRINT "YES"
```

Color Basic uses a lot of user RAM in a "4K" machine. When the system is first turned on, typing PRINT MEM yields only 2343. Typing CLEAR 0:PRINT MEM produces 2543. Evidently 200 bytes are automatically cleared for string variables on power-up. There is no convenient way to find out how much string space is available (there is no FRE command).

DEFINT, DEFSNG, DEFDBL, and DEFSTR are all absent. If a variable (one or two characters, the first of which is a letter, the second either a letter or number) is followed by a "$" it is a string variable, otherwise it is a numeric variable. Numeric variables are stored as 5 byte floating point numbers (compared with 4 bytes for single precision and 8 bytes for double precision in Level II). This results in 9-digit accuracy whether you need it or not.

Also missing are: VARPTR, STRING$, COS, ATN, LOG, EXP, the up-arrow which allows raising a number to a power in Level II, and the ON ERROR GOTO instruction.

So much for the drawbacks.

Further Facts

Holding the shift-key while simultaneously pressing either up-arrow, down-arrow, right-arrow, or CLEAR produces a left arrow (!?), left bracket, right bracket, or backslash respectively.

The cursor is constantly cycling through the eight colors. Some people may find this annoying.

Screen memory begins at location 9216 (decimal) and continues for a total of 512 locations. Surprisingly, it also begins at location 25600. PEEKs from and POKEs into either block of memory produce identical results. Figure 1 is a program which POKEs the entire Color Basic character set onto the screen.

Figure 1

```
10 S=9216+64' 2ND LINE OF SCREEN
20 CLS:FOR I=0 TO 255
30 POKE S+I,I:NEXT
```

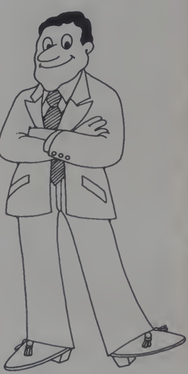
TRS-80 Color Computer

CLEAN UP YOUR ACT

Professionals demand quality products tailored to meet their needs. Super-Text continues to provide affordable word processing while meeting the highest performance standards. Now Muse adds two new products to our family of business oriented software.

The Form Letter Module joins Super-Text to perform a broad range of sophisticated word processing jobs. Have all the features a serious businessperson wants without the prohibitive cost of larger systems. And by not being a single purpose computer, your Apple and Muse software can help solve many other business problems.

Take Data/Plot for example. After increasing sales with Super-Text form letters, you may need attractive graphs and charts to show your boss just how you've increased profits as well as your salary which explains your $800 designer suits. After reviewing the brief specs below, visit your nearest computer retailer who carries Muse and Apple products and ask for a demonstration of our business software.



Businessware for Professionals

Using Super-Text ($150) as your word processing editor, the Form Letter Module ($100) allows you to create complex form letters which can be customized for each person or business entered in the Address Book Mailing List ($50). On screen prompts and menus facilitate operator entry of additional information and allow piecing together of complex documents. Runs on the Apple II or II plus with 48k and disk drive.

Data Plot's easy editing features allow you to create and modify a wide variety of full color graphic representations of numerical information. Bar charts, including additive bars, as well as single and multiple line charts may be plotted individually or cumulatively. Pie charts are easily sliced. All figures may be output to a graphics printer. Runs on an Apple II or II plus with 48k, Applesoft ROM and disk drive.

MUSE SOFTWARE™

330 N. CHARLES STREET
BALTIMORE, MD 21201
(301) 659-7212

CIRCLE 225 ON READER SERVICE CARD

TRS-80 Color, continued...

The results show that 0 through 127 are upper and lowercase characters (lowercase is shown as reverse-video uppercase, but will appear as lowercase on a printer), while 128 through 255 are graphic characters—16 graphic characters for each of the eight colors:

Green: 128-143, Yellow: 144-159, Blue: 160-175, Red: 176-191, Buff: 192-207, Cyan: 208-223, Magenta: 224-239, and Orange: 240-255.

The first number in each group 128, 144, ..., 240) has the pixel in the lower-right corner of the square lit and so on. To determine the code for a given combination of lit pixels, start with the first number in the group corresponding to the color you wish to use (for example, use 160 for blue) then add the numbers in the pixels you wish to light (see figure 2).

Figure 2

```
8  4

2  1
```

Thus a graphics character which is blue on the top half and black (unlit) on the bottom half can be PRINTed by using (CHR$(172) (160 for blue plus 8 and 4 for the desired top pixels).

Several Color Basic statements are extensions of their Level II counterparts. The SET statement still turns on a graphics point, but the command now contains three quantities in parentheses instead of two: SET(H,V,C). H is the horizontal coordinate and may range from 0 to 63; V is the vertical coordinate any may range from 0 to 31; C is the color of the point and my be any number from 0 to 8, each of which represents a different color. Thus the resolution of the screen is 64 x 32 or 2048. This is one-third the resolution of the Level II screen. As noted earlier, there are four pixels in each PRINT @ position. All "lit" pixels in a given PRINT @ position must be the same color. If you SET one of them green, and later SET another one red, the pixel which was SET first will also become red.

CLS, the command used to clear the screen, may be followed by a number from 0 to 8. Each number clears the screen to a different color. (CLS 9 clears the screen to the default color green and then prints the word "MICROSOFT"!).

POINT(X,Y) returns a value from 0 to 8 depending on the color of the pixel at X,y. If a character is present at the PRINT @ position containing X,Y then POINT(X,Y) will equal -1.

CLEAR can be followed by one or two quantities. The first, as in Level II, sets aside string space. The second (if used) is the highest address Basic can use (similar to MEMORY SIZE or MEM SIZE in Level II, but can now be easily altered without having to simulate power-up conditions.)

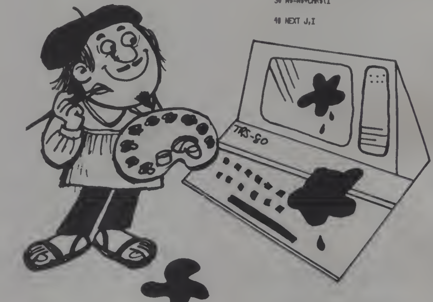
CLOAD functions the same as it does in Level II, but will accept program names up to eight characters long.

CLOADM loads a machine language program from tape., If a number appears after the filename as in:

```
CLOADM"PROC",1000
```

the normal loading address will offset by that number.

Level II programmers can say goodby to machine language sound routines. The SOUND P,T statement can produce a note within a range of four full octaves (as P varies from 1 to 255) while T controls the duration in increments of six hundredths (.06) of a second. An appendix in *Getting Started With Color Basic* shows values to use for P to obtain any of the notes contained in the four octaves.



A Question Of Speed

As with Level II, there are ways to speed up your graphics (although none are mentioned in either manual). These mehods are similar to their Level II counterparts: POKE graphics and string graphics. In the Color Computer Operation Manual there is a Color Adjustment Test program which draws 8 vertical color bars. This is to allow you to adjust your television so that it will display the proper colors. The program uses SET graphics and takes over 33 seconds to fill the screen (No listing of the program is given here in order to comply with a rather ferocious warning in the front of the manual against reproducing any of the contents; however, I think I'm safe in passing along the information that it consists of 6 short lines.) Figure 3 gives a routine to display the same 8 color bars, but using POKE graphics instead of SET. This cuts the time in half (about 16 seconds).

Figure 3

```
10 FOR ROW=0 TO 15
20 FOR COL=0 TO 31
30 P=143+16*INT(COL/4)
40 POKE 9216+32*ROW+COL,P
50 NEXT:NEXT
60 GOTO 60
```

Figure 4 is a program to do essentially the same thing by creating and PRINTing the appropriate string several times. This method takes about two-thirds of a second!

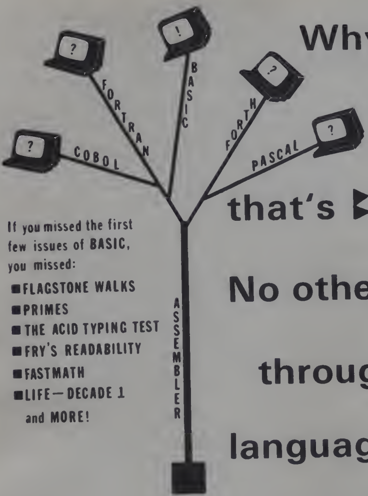
Figure 4

```
10 FOR I=143 TO 255 STEP 16
20 FOR J=1 TO 4
30 A$=A$+CHR$(I
40 NEXT J,I
50 CLS0:FOR I=1 TO 15:PRINT A$;:NEXT
60 GOTO 60
```

Cassette Operation

Both manuals strongly recommend the use of Radio Shack's CTR-80A recorder for use with this computer. The first thing I did was attempt to connect my ancient CTR-41 (which came with my original 4K Level I TRS-80) to the Color Computer. No luck. The jack on the back of the computer is slightly smaller than the DIN connector. Therefore the following information has not been verified.

CSAVE and CLOAD operate 1500 baud; three times faster than Level II. Files may be given names up to eight characters long. The SKIPF "NAME" instruction causes the tape to be advanced to the end of a program called NAME. There are commands to turn the MOTOR ON and MOTOR OFF, and there is a command called AUDIO which is supposed to send the sound on a cassette to your television



Why not subscribe to the magazine that's BASICally for you?

No other magazine cuts through the forest of languages and gives you what you want: BASIC!

If you missed the first few issues of BASIC, you missed:

- FLAGSTONE WALKS
- PRIMES
- THE ACID TYPING TEST
- FRY'S READABILITY
- FASTMATH
- LIFE—DECADE 1

and MORE!

SUBSCRIBE NOW! MAKE BASIC YOUR ☆☆☆☆☆ FAVORITE COMPUTER MAGAZINE

YES, I LOVE BASIC!
— START MY SUBSCRIPTION WITH THE NEXT ISSUE

I ENCLOSE ______ FOR 12 ISSUES

Sample issue $2.75 ______

BASIC MAGAZINE
SOUTH RIVER NEW JERSEY 08882

____
Name (Please Print)

____
Street

____
City State Zip

I may cancel my subscription any time and receive a refund for the unused balance

Subscription Rates: $20.00 USA, $24.00 Elsewhere
Airmail add $10.00 US Dollars only

I Would like to see these features in BASIC:

Games ___ Business ___ Science ___

Software Reviews ___ Hardware Reviews ___

New Products ___ Other ______________

I am NOT interested in ______________

____

"THE MAGAZINE FOR COMPUTERISTS"
A DIVISION OF SAFETY-SWEET, INC.

TRS-80 Color, continued...

speaker. This offers the possibility of combining music, speech, etc. with Basic programs.

Program And Other Storage

(The following information is of a more technical nature. It was discovered by this reviewer and is subject to change pending the observations of others.)

The program in Figure 5 will list all the keywords in Color Basic, along with their one- or two-byte decimal tokens.

Figure 5

```
2000 X=128:PRINT X;

2010 FOR I=43622 TO I+255:M=PEEK(I)

2020 IF M<128 THEN PRINT CHR$(M);:GOTO 2050

2030 PRINT CHR$(M-128),:X=X+1:IF X<181 THEN PRINT X;:GOTO 2050

2040 PRINT " 255";X-53;

2050 NEXT
```

Table 1 gives the decimal token values of the Color Basic instructions. If there are two words for a given number, the second word (usually a function, which means: immediately followed by one or more items in parentheses) has a two-byte token: the first byte is 255 and the second byte is the same as that for the first instruction. For example, the instruction FOR is stored as 128 while the SGN function is stored as 255 followed by 128. There are many instructions which are familiar friends from Level II (and even a few reminiscent of Disk Basic such as OPEN and CLOSE). Some instructions are sadly absent (the loss of STRING$ and VARPTR grieved me the most) and some new instructions have been introduced.

Table 1

128 FOR	SGN	146 POKE	INKEY$	164 TAB(
129 GO	INT	147 CONT	MEM	165 TO
130 REM	ABS	148 LIST		166 SUB
131 '	USR	149 CLEAR		167 THEN
132 ELSE	RND	150 NEW		168 NOT
133 IF	SIN	151 CLOAD		169 STEP
134 DATA	PEEK	152 CSAVE		170 OFF
135 PRINT	LEN	153 OPEN		171 +
136 ON	STR$	154 CLOSE		172 -
137 INPUT	VAL	155 LLIST		173 *
138 END	ASC	156 SET		174 /
139 NEXT	CHR$	157 RESET		175 (up-arrow)
140 DIM	EOF	158 CLS		176 AND
141 READ	JOYSTK	159 MOTOR		177 THEN
142 RUN	LEFT$	160 SOUND		178 >
143 RESTORE	RIGHT$	161 AUDIO		179 =
144 RETURN	MID$	162 EXEC		180 <
145 STOP	POINT	163 SKIPF		

This reviewer could find no mention of the word SUB (token 166) in either manual.

Going Deeper

The following memory locations are in decimal. There are four two-byte pointers starting at locations 25 and 26 and ending with locations 32 and 32. With one surprising difference, these pointers appear to serve the same purpose as those at 16548, 16623, 16625, and 16627 in Level II. The first of these contains the address of the beginning of the Basic program area. The second pointer indicates both the end of the Basic program and the beginning of simple (unsubscripted) variables. The fourth pointer gives the end of array variables and the beginning of free memory.

The surprising difference is that each of these 2-byte pointers is stored most-significant first, least-significant-byte second—just the opposite of Level II. For example, in location 25 and 26 we find the numbers 6 and 1 respectively. Basic programs start at location 6 x 256 +1 or 1537.

Except for this difference, Basic programs are stored in essentially the same internal format as they are in Level II. For example, if the one line program

```
10 REM
```

is entered, the following will be found from locations 1537 to 1544:

```
6, 7, 0, 10, 130, 0, 0, 0
```

The first two bytes, when interpreted as a pointer (again in the order MSB, LSB) give 1543. This is the address of the beginning of the next line, if there is one, or a pointer to two consecutive zeros, as in this case, to indicate the end of the program.

The next two bytes, 0,10, are the line number: 256 x 0 + 10 = 10.

130 is the token for REM. The next byte (0) signals the end of the current line.

Immediately following the program, the simple variables are stored. Numeric and string variables each take seven bytes. A numeric variable such as G1 is stored as:

```
N1, N2, B1, B2, B3, B4, B5
```

Where N1 is the ASCII value of the first letter of the name (ASC("G") = 71 in this case), N2 is the ASCII value of the second character (ASC("1") = 49) if there is one, or zero in the case of a one-letter name. B1 through B5 are the floating point representing of the actual value of the variable. A discussion of floating point numbers is beyond the scope of this review.

String variables are stored a little differently:

```
S1, S2, LEN, U1, L1, L2, U2
```

While S1 is again the ASCII value of the first letter of the string variable name, S2 is the ASCII value of the second character increased by 128. LEN is the length of the string, L1 and L2 comprise the pointer to the string itself. As in Level II, this may be either a location in the Basic program (if the string is defined simply as in: A$= "HELLO ") or an address in string storage space (if the string definition requires any string minipulations as in: A$="HELLO" +"THERE")

What are U1 and U2? I don't know. Both locations were zero in all strings I examined.

Summary

In conclusion, the TRS-80 Color Computer is an excellent computer for the under $400.00 price. You will probably decide to add a recorder and upgrade to 16K of memory if you plan to do any serious programming.

Radio Shack has incorporated many of the features which Level II users have requested in the past. Since all the electronics for interfacing printers, RS-232-C devices, etc. are built in, there should be no need for a cumbersome expansion interface in the future. With lowercase printing capability as standard equipment, another complaint of earlier Level II users has been addressed.

My wife and I were sorry when it was time to return the computer to work because we enjoyed playing with it so much. □

"And in what manner did the defendant strike you after you placed him in checkmate?"

After Conquering Britain, Commodore Sets Sail for the Colonies

No PET Peeves

Peter Fee

Commodore wants you. That in itself is not news; Apple, Atari and Radio Shack want you, too. What *is* news is that Commodore now feels that it has the means to re-establish its lead in the American market—and it wants that more than anything. After a few years of taking a back seat to the leaner, meaner competition, Commodore is about to introduce its new home computer—the VIC 20.

This past September 30th I attended a rather hastily called press conference at New York's "fashionable" Four Seasons, hoping to get a firsthand look at the VIC. Normally one would expect a Commodore conference to be attended almost exclusively by the computing press; instead, I felt as if I'd walked onto the floor of the Stock Exchange. Though the meeting had ostensibly been called to announce Commodore's new line-up, there was another reason behind it. Commodore stock had been (and still is) going right through the Wall Street roof. The Securities and Exchange Commission was rumored to be very interested in this development (unlike most rumors, this one turned out to be true) and so, the main purpose of the conference was to show key security analysts that there was a reason for all the excitement.

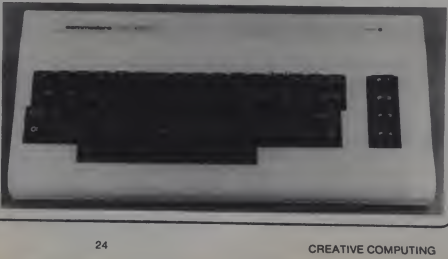
Apart from the introduction of the new equipment (Oh yes, I *will* be getting to that), Commodore also talked about their plans to back it up. Among the artillery: an advertising budget of between five and eight million dollars and a pledge to improve the network of 500 or so PET dealers by providing them with more service and software support.

When Commodore talks about the competition, they speak less of Apple and Radio Shack than of the Japanese companies they believe will try to dominate the market. Commodore has decided to take the offensive, introducing the VIC 20 in Japan as the VIC 1001 (pictured here—note the Japanese characters on the keyboard). Apparently Commodore seems determined to learn the lesson GM, Ford and Chrysler learned too late; the only "K" Commodore has to push is in the VIC 20.

But enough of that; let's get to the goodies. The VIC 20 offers: color, sound, programmable function keys, 5K RAM with memory expansion to 32K, 6502 microprocessor, full-size typewriter keyboard, external expansion ports, 22 character by 23 line screen display, high resolution graphics, graphics character set, external plug-in memory and program cartridges, joystick/paddles/lightpen and standard PET Basic. Taken separately, none of these features are particularly new; but put them together, and you haven't got a bad little machine.

In case you're wondering, VIC stands for Video Interface Chip, which incorporates RAM, ROM and video control circuitry all on the same chip. It was the development of this chip that allowed Commodore to reduce the total number of components needed to build the VIC 20.

But what does this have to do with the price of computers? Everything, and more. When the number of components is reduced, the price is also reduced, and you have a revolution on your hands. The VIC 20 will sell for only $299; how long before an equivalent computer sells for under $200? At the VIC's price, the home computer can be to the American family of the 1980's what the television set was to the American family of the 1970's (without the mind-numbing effects, of course. You won't be able to watch *I Love Lucy* reruns on your PET). A high quality machine at a low price will change things so fast you might miss it even if you *don't* blink.



The most important question here is: Is the VIC 20 a high-quality machine at a low price? Well, $299 is an *incredibly* low price. As for the high quality, I can't answer that yet. Though I did see the VIC in action at the Commodore press conference, I never got the chance to play with it myself. An Apple, an Atari, and a TRS-80 were on hand "to show the superiority of the VIC 20," but unfortunately the promised comparison never materialized. The VIC will be formally introduced at the Consumer Electronics Show in Las Vegas in early January, and will go on sale immediately thereafter.

The VIC is not all Commodore has to offer. A new single floppy disk unit, the CBM 2031, is being introduced, with a serial-bus version for the VIC to follow. Both the CBM 2031 and the 2031S will sell for under $600.

Commodore will also be marketing the CBM 8096, a 96K version of the 8032 80-column business computer. This expansion in internal storage capacity would double the amount of RAM in a 32K computer and permit large programs to reside in and cycle through the expanded memory space. The price of the 8096 has not yet been announced.

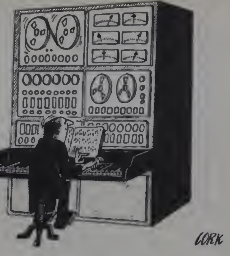
The Wordcraft 80, Commodore's new wordprocessing system, could establish the company in that field. The Wordcraft 80, combined with a Commodore business system, would sell for about $5000, another price breakthrough.

Commodore's big suprise is its line of cash register computer systems, which combine computerized business record keeping with conventional cash register devices. Three separate systems have been developed: Registers for 1) grocery and cost-plus stores, 2) restaurants, and 3) retail inventory control. Each system includes a small computer with built-in CRT display screen, receipt printer, cash draw electronically-keyed to the computer, and optional floppy disk storage unit.

A high quality machine at a low price will change things so fast you might miss it even if you don't blink.

I saw the demonstration of the restaurant cash register system, and I just fell in love with it. This register does everything but tell you that the prime rib is a little fatty tonight. Using this system, a server can call up the menu list, place the customer's food and drink orders, and specify such variables as the salad dressing, a la carte items and meat preparation. After the meal, the bill can be tallied line by line for each expenditure by calling up the table number, waitress number, or ticket number. The transaction amount and amount paid are entered, and change due displayed on the screen. Total information for each order is stored on a floppy disk for subsequent data retrieval and data processing. All in all, very nice.

The question remains: Can this new equipment re-establish Commodore in America? From the standpoint of price, the answer is most definitely "yes." Commodore has shown great interest in making the home computer affordable to the middle class, as opposed to the "Let them eat cake" attitude of some companies. Investor interest also cannot be denied, as the SEC will tell you. (By the way, no irregularities were found in the matter.) The quality of the equipment is the only unanswered question, but since everything I've mentioned here is sheduled for a January 1981 release, the question won't stay unanswered for long. We'll keep you posted. □



Atari Music Composer Cartridge

Leonard Nasman

Do you hear music inside your head, but can't play an instrument well enough to get it out? Atari has put together a tool for all of you budding Beethovens that makes musical compositions easy to create, modify, and play. The Music Composer Cartridge simply plugs into an Atari 400 or 800 ROM slot and allows a user to enter, rearrange, play, and save up to four-part harmony.

Those that are familiar with the Atari system know it to be a powerful personal computer that comes with great graphics, an expanded character set, and built-in tone generators. Atari has now added a cartridge that makes music entry easy for those who are not programming experts.

When you plug the cartridge in and close the lid the first of five menus appears on the TV screen (Atari attaches to a standard color TV with an antenna switch box similar to their video games). This first "main menu" lists; Edit music, Arrange music, Save, Retrieve, DOS, and Listen. Typing the first letter of any of the first four of these, followed by the "return" key, causes a jump to one of the four sub menus, DOS causes the disk operating system menu to be displayed (if no disk drive is attached, it will jump to "Atari memo pad"). "Listen" causes whatever music is stored to be played.

The "Edit" sub menu allows you to check measures, enter notes or change the meter, key signature or tempo. The notes are entered by selecting a phrase number (O to 9), selecting a measure within a phrase, and typing the desired note code. Typing "C4Q" will get a C in octave 4 played as a quarter note. Typing "BF3S" will get a B flat in the third octave played as a dotted sixteenth. Typing "FS5E" will get an F sharp in octave 5 played as an eighth note. "RH" will yield a half note rest. If you can associate notes with their letter names, they are quite easy to enter.

Once you have entered the several measures of notes that constitute a musical phrase (any number of measures can make a phrase), you jump back to the main menu and decide what to do next. If you elect to "Arrange" by typing "A" followed by the return key, the Arrange menu will be displayed. This menu provides eight options: Count, Display, Go to line, Show the arrange menu, Play phrase, Stop (return to main menu), Transpose, and Volume. Your arrangement is limited to 20 lines of these options. You can arrange up to four different voices that will play together when you enter "L" for listen.

To enter a song like the old round "White Coral Bells", you would proceed as follows:

1. Type "E" (edit)
2. Type "P" (phrase)
3. Enter phrase number "1"
4. When the prompt "Erase?" appears on the screen type "Y" (since this is a new selection we want to clear the phrase, if we only want to change something already entered we would type "N").

At this point the screen will show bass and treble cleffs with key signature, meter, and a cursor, below which will be printed:

PHRASE 1
MEASURE 1
10232 FREE
L (A) O D (.) (T)
SM1, NOTE?

The first two lines show the phrase and measure you are in. The number in fron front of "FREE" shows how much space is left in memory. The fourth line shows the format for entering notes: L-Letter, A-Accidental, O-Octave, D-Duration, (.)-Dot, T-Tie. "SMI" in the last line is a mini menu that reminds you that you can opt to "S"-Stop (return to the edit menu), "M" select a measure, or "I" insert a measure, as well as enter a note at this point. Now back to "White Coral Bells".

5. Enter notes in measures and phrases. For this song let "White cor-al/bells up-/on a slen-der/stalk" be four measures in phrase one, and "Lil-ies of the/val-ley deck my/gar-den/walk" be four measures of phrase two. Also, enter four measures of rest as phrase three.
6. After you have entered these three phrases, type "S" to return to the edit menu, then "S" again to return to the main menu.
7. Now type "A" for arrange. The prompt "VOICE#?" appears. Enter "1" for the first voice. Line #1 shows "DISPLAY", and line #2 shows "PLAY PHRASE" automatically. The cursor location is identified by coloring one line blue instead of the normal white letters.
8. Arrange the four voices. To play a four part round they should look like this:

VOICE #1	VOICE # 2
PLAY PHRASE 1	PLAY PHRASE 3
PLAY PHRASE 2	PLAY PHRASE 1
PLAY PHRASE 1	PLAY PHRASE 2
PLAY PHRASE 2	PLAY PHRASE 2
	PLAY PHRASE 2

VOICE #3	VOICE #4
PLAY PHRASE 3	PLAY PHRASE 3
PLAY PHRASE 3	PLAY PHRASE 3
PLAY PHRASE 1	PLAY PHRASE 3
PLAY PHRASE 2	PLAY PHRASE 1
PLAY PHRASE 1	PLAY PHRASE 2
PLAY PHRASE 2	PLAY PHRASE 1
	PLAY PHRASE 2

9. Type "L" and listen to the result.

To simulate a soprano and bass voice in the arrangement, insert "T", "12" in voice #2, and "T", "-12" in voice #3 (12 half steps equals a full octave).

To save the composition, jump to the "SAVE" menu where you have the choice of saving everything, any or all phrases, or any or all voice arrangements on cassette or disk. The "RETRIEVE" menu allows the same options.

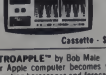
Creating your own composition is as simple as entering and manipulating phrases. Now if Atari would only add the kind of synthesizers found in electronic organs, about 32 voices, and around 200 lines for arranging, we could really compete with Bach. In the meantime, the Atari Music Composer Cartridge will be helping budding composers polish their creative skills. □

Leonard Nasman, 3867 Braidwood Dr., Columbus, OH 43220.

Software for the Apple II and Apple II Plus*

ASTEROIDS IN SPACE™
By Bruce Wallace
An exciting space action game! Your space ship is traveling in the middle of a shower of asteroids. Blast the asteroids with lasers, but beware — big asteroids fragment into small asteroids! The Apple game paddles allow you to rotate your space ship, fire its laser gun, and give it thrust to propel it through endless space. From time to time you will encounter an alien space ship whose mission is to destroy you, so you'd better destroy it first! High resolution graphics and sound effects add to the arcade like excitement that this program generates. Runs on any Apple II with at least 32K of RAM and one disk drive.

On Diskette - $19.95

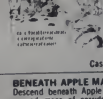


FASTGAMMON™ By Bob Christiansen.
Sound, hi res, color, and musical cartoons have helped make this the most popular backgammon playing game for the Apple II. But don't let these entertaining features fool you — FASTGAMMON plays serious backgammon. Runs on any Apple II with at least 24K of RAM.

Cassette - $19.95 Diskette - $24.95

ASTROAPPLE™ by Bob Male
Your Apple computer becomes your astrologer, generating horoscopes and forecasts based on the computed positions of the heavenly bodies. This program offers a delightful and stimulating way to entertain friends. ASTROAPPLE produces natal horoscopes (birth charts) for each person based on his or her birth data. Any two people may be compared for physical, emotional, and intellectual compatibility. The program is written in Applesoft BASIC with machine language subroutines. It requires either RAM or ROM Applesoft and at least 32K of memory.

Cassette - $14.95 Diskette - $19.95



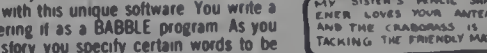
FRACAS™ by Stuart Smith.
A fantastic adventure game like no other! Up to eight players can participate in FRACAS at the same time. Journey in the land of FAROPH, searching for hidden treasure while warding off all sorts of unfriendly and dangerous creatures. You and your friends can compete with each other or you can join forces and gang up on the monsters. Your location is presented graphically and sound effects enliven the battles. Save your adventure on diskette or cassette and continue it at some other time. Both integer BASIC and Applesoft versions included. Requires at least 32K of RAM.

Cassette - $19.95 Diskette - $24.95

BENEATH APPLE MANOR™ by Don Worth.
Descend beneath Apple Manor into an underground maze of corridors, rooms, and secret passages in quest of rich and powerful treasures. The dungeon complex consists of many floors, each lower level more dangerous than the last. If you can reach the lowest level, you may even find the ultimate object of your quest, the fabled golden apple of Apple Manor. Strategy is extremely important as you deal with a variety of monsters, each with its own characteristics. Written in integer BASIC with machine language subroutines. Requires integer BASIC and at least 16K of RAM on cassette or 32K of RAM on diskette.

Cassette - $14.95 Diskette - $19.95

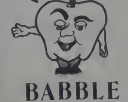
BATTLESHIP COMMANDER™ by Erik Kilk and Matthew Jew



A game of strategy. You and the computer each start out by positioning five ships of different sizes on a ten by ten grid. Then the shooting starts. Place your volleys skillfully — a combination of logic and luck are required to beat the computer. Cartoons show the ships sinking and announce the winner. Sound effects and flashing lights also add to the enjoyment of the game. Both Applesoft and integer BASIC versions are included. Requires at least 32K of RAM.

Cassette - $14.95 Diskette - $19.95

BABBLE™ by Don Worth
Have fun with this unique software. You write a story, entering it as a BABBLE program. As you write the story you specify certain words to be selected by the computer or entered from the keyboard at execution time. Run the program and watch BABBLE convert your story into an often hilarious collection of incongruities. The ways in which BABBLE can entertain you are limited only to your imagination. You can compose an impressive political speech or write poetry. You can plan a dinner menu. You can even form images on the screen or compose musical tunes with the help of BABBLE. The cassette version requires at least 16K of RAM and the diskette version requires at least 32K of RAM. BABBLE is written in machine language and runs on any Apple II computer.



BABBLE

Cassette - $19.95 Diskette - $24.95

L I N K E R

LINKER by Don Worth
Turn your Apple II or Apple II Plus into a powerful and productive software development machine with this superb linking loader/editor package. LINKER does the following and much more.

- Dynamically loads and relocates suitably prepared machine language programs anywhere in RAM.
- Combines a main program with subroutines. You can assemble a subroutine once and then use it with as many main programs as you wish.
- Produces a map of all loaded routines, giving their location and the total length of the resulting module.
- Contains a library of subroutines including binary multiplication and division, print text strings, delay, tone generator, and random number generator.

Linker works with virtually any assembler for the Apple II. Requires 32K of RAM and one disk drive.

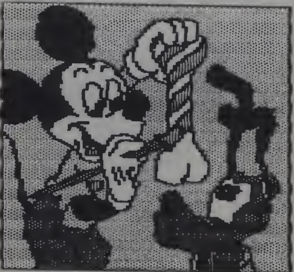
Diskette - $49.95
Manual Only - $19.95

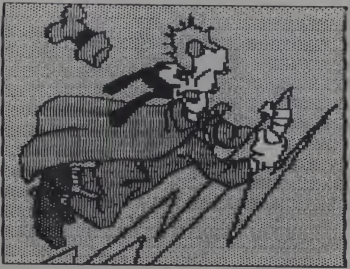
QUALITY SOFTWARE
6660 Reseda Blvd., Suite 105, Reseda, CA 91335
(213) 344-6599

Now exclusive distributors for products from The Software Factory, Newhall, California
*Apple II and Apple II Plus are trademarks of Apple Computer, Inc.

WHERE TO GET IT: Call us at (213) 344-6599 for the name of the Quality Software dealer nearest you. If necessary you may order directly from us. Mastercard and Visa cardholders may place orders by telephone. Or mail your check or bankcard number to Quality Software, 6660 Reseda Blvd., Suite 105, Reseda, CA 91335. California residents add 6% sales tax. SHIPPING CHARGES: Within North America orders must include $1.50 for first class shipping and handling. Outside North America the charge for airmail shipping and handling is $5.00. Pay in U.S. currency.

CIRCLE 231 ON READER SERVICE CARD





The Apple Graphics Tablet

George Sternecker

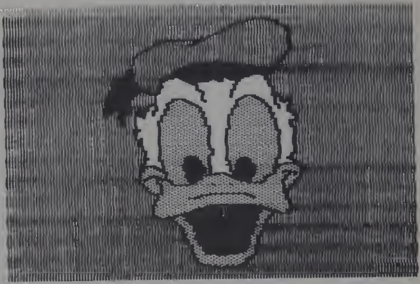
The high resolution color capabilities of the Apple computer remain largely unexploited. Most software programs do not take advantage of the attention grabbing affects of hi-res color graphics. There is no excuse for this, since the hi-res screens can be easily utilized by various graphics tablets on the maket. Here, we will focus upon the Apple Graphics Tablet.

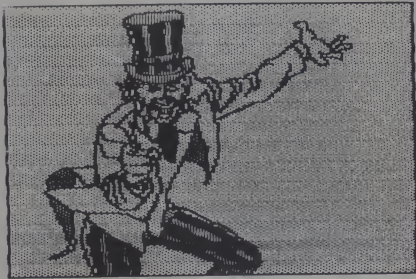
The Graphics Tablet is a magnetic bit pad which digitizes signals created by touching a magnetic pen to the tablet surface. The AGT consists of the bit pad itself, an overlay grid, magnetic pen, interface card, software, manual, and an anti-static cloth. The card goes into any peripheral slot in the computer (normally slot 5), and the bit pad and pen are both connected by wires to the interface card.

The tablet is 15"x15", and matches the Apple computer in color and styling. The AGT draws on HGR2, the second high resolution page, and uses the standard hi-res colors—black, white, blue, orange, green and violet. On the tablet itself is a mylar grid sheet which is aligned by using the calibration program. Every command on the AGT is displayed across the top of the grid sheet. Commands and program include reset, clear screen, pen color, recall and store picture, background color, area/distance, frames, boxes, straight lines, window, viewport, etc. The magnetic pen is used to draw on the screen, and also to execute any of these commands. The computer keyboard is used only to specify picture names and the disk drive numbers. One seldom has to leave the tablet surface while using the AGT.

The AGT is easy enough for anyone, including the computer novice, to use. The computer only "draws" when the pen

Graphics by Bob Bishop of Apple Computer.





is actually touching the surface of the AGT. When the pen is close to the surface, crosshairs appear on the video screen to indicate location. Manipulating the pen is not much harder than writing with an ordinary pen, and with a little practice, you can write cursive on the video screen with panache.

Since the AGT operates on magnetic principles, static electricity is its enemy. If the user has a static charge, or if there is a bad electrical ground, strange anomalies will appear on the screen during the drawing mode. When creating graphics, always save the picture from time to time, so that if something goes wrong, all is not lost.

Two interesting commands on the AGT are "window" and "viewport". WINDOW allows you to specify an area on the grid, and have that area equal the entire video screen. That is, a square inch could be specified as the window, and if a line were drawn across that square inch, it would extend across the entire video screen. VIEWPORT allows you to isolate a section of the grid for drawing, thus freezing the remaining portion of the hi-res screen to avoid accidentally drawing there. This feature is good for editing or for fine detail work on the screen. The AGT *will not* shrink or enlarge what is already on the hi-res screen. You can also draw a rectangle, either in outline or filled, just by specifying the two end points of one diagonal. Graphics created with the AGT can be inserted into programs by going to HGR2 and executing a BLOAD PIC (picture name).

AGT operates on Applesoft basic and requires at least one disk drive. The suggested price is $795.00, making the AGT one of the more expensive graphics tablets available. (Apple Computer Inc., 10260 Bandley Drive, Cupertino, CA 95014) □

The Casheab Music Synthesizer

Jon Bondy

While at the West Coast Computer Faire in March 1980, I discovered the Casheab music synthesizer for the first time. Previous to this, the best music synthesizing equipment one could purchase for use on the S-100 bus was the Solid State Music SB1 board, a board with distinct limitations. Also, at that time, the only music synthesis boards for the Apple were made by ALF, and they only produced square waves. By comparison, the Casheab was extremely versatile and reasonably priced, so I purchased one of their first units, receiving it in June.

To give you an idea of how far things have come in about three years, the SSM SB1 costs about $150 (kit) and will synthesize one voice with 32 8-bit samples per cycle of the waveform and 15 steps of amplitude control. The Casheab costs about $1000, but provides 32 voices with sixteen waveforms each with 1024 12-bit samples per waveform and 255 levels of amplitude control. Although more expensive initially, the Casheab is far less expensive for someone who is serious about creating multi-voice music. In addition, the Casheab has FM capabilities, allowing it to do vibrato and more complex FM synthesis, as discussed below.

The standard Casheab synthesizer consists of two S-100 boards linked together by a ribbon cable: one a controller, the other one the synthesizer itself. The former contains the processor interface (S-100), timing generators, an accumulator, and the digital-to-analog section. The latter contains the frequency-generation hardware, waveform memories, and amplitude-control hardware. The synthesizer board contains a 16-MHz bit-serial signal processor which scans through the waveform memories at a rate determined by the frequency-generation hardware, to produce amplitude samples at a fixed rate through a time-multiplexing scheme. Because of the ribbon cable, the synthesizer cannot be debugged completely without having two extender boards; however, you can debug each board individually for many problems, by removing the jumper. Due to the high frequencies used, the boards are NOT available as kits, only as assembled and tested units.

The Casheab has FM capabilities allowing it to do vibrato and more complex FM synthesis.

The synthesizer has so many control parameters that it is memory-mapped in order to avoid tying up most of the I/O ports. It uses 256 bytes of memory for control, usually allocated at ϕF8ϕϕH, although I use ϕFFϕϕH. Because all of the memory locations used for synthesizer control are write-only, you can run this board at the same address as a working memory board and the two will not interfere with each other; when the synthesizer is not needed, that area will look like regular memory, and when the memory is not needed, the synthesizer will be available. The synthesizer does assert the wait lines to allow for internal synchronization, which could cause the regular memory to appear to be slower with the synthesizer in the computer. Inadvertently writing to the memory when the synthesizer is running will have some fairly discordant effects, however.

Each of the 32 channels has a two-byte Frequency Control Word (FCW) which controls the rate at which the waveform for the particular channel is scanned. Frequencies can be specified in multiples of approximately 0.3 Hertz from 0 Hertz to about 19 KHz, which provides reasonably precise control for musical purposes. Each channel also has what Casheab calls a 'weight', but which I call an Amplitude Control Word (ACW). These allow each channel to have amplitudes from zero (off) to 255. Each channel also has one byte for selecting which of the sixteen waveform tables it is to use, and a byte to indicate whether it is to FM-modulate the channel two above it or not. An FM channel, thus specified, uses its output to increase or decrease the rate at which the channel two above it is scanned, thus increasing or decreasing the pitch of that note. A channel which is used for FM is not heard at the synthesizer output; a non-FM channel is summed with all other non-FM channels and their sum is available at the sound output of the synthesizer for direct connection to a music amplifier. The synthesizer produces a single channel of audio output, combining all 32 channels into one signal.

The waveform tables are loaded by loading a special byte in the memory map with the number of the waveform table to be loaded, and then loading the table data sequentially into another special byte location in the map. One additional special memory location is used for overall scaling of the synthesizer, since the output with all 32 channels in use is significantly greater than with only one channel.

The board is strewn with wire-wrappable jumpers, to allow the user to re-configure it for either 4 or 16 waveforms, either 1024 or 2048 samples per waveform, and either 10,16, or 32 channels (yielding sampling rates of 50, 34, or 17 KHZ (for frequency responses of 25, 17, or 8.5 KHZ, respectively). Also, either normal or inverted phase one or phase two S-100 bus clocks can be used to trigger the board, allowing use with all 'standard' processor boards.

The first thing which impressed me about the synthesizer was the care which went into it, in terms of both the quality of the documentation and the boards themselves.

Jon Bondy, Box 148, Ardmore, PA 19003.

DYNACOMP

Quality software for:

ATARI	TRS-80 (Level II)
PET	NORTH STAR
APPLE II Plus	CP/M 8" Disk

GAMES

BRIDGE 2.0 (Available for all computers) — Price: $17.95 Cassette, $21.95 Diskette

An all-inclusive version of this most popular of card games. This program both BIDS and PLAYS either contract or duplicate bridge. Depending on the contract, your computer opponents will either play the offense OR defense. If you bid too high, the computer will double your contract! BRIDGE 2.0 provides challenging entertainment for advanced players and is an excellent learning tool for the bridge novice.

HEARTS 1.5 (Available for all computers) — Price: $14.95 Cassette, $18.95 Diskette

An exciting and entertaining computer version of this popular card game. Hearts is a trick-oriented game in which the purpose is not to take any hearts or the queen of spades. Play against two computer opponents who are armed with hard-to-beat playing strategies.

CRIBBAGE 2.0 (TRS-80 only) — Price: $14.95 Cassette, $18.95 Diskette

This is a well-designed and nicely executed two-handed version of the classic card game, cribbage. It is an excellent program for the cribbage player in search of a worthy opponent as well as the beginner wishing to learn the game, in particular the scoring and jargon. The standard cribbage score board is continually shown at the top of the display (utilizing the TRS-80's graphics capabilities), with the cards shown underneath. The computer automatically scores and also announces the points using the traditional phrases.

CHESS MASTER (North Star and TRS-80 only) — Price: $19.95 Cassette, $23.95 Diskette

This complete and very powerful program provides five levels of play. It includes castling, en passant captures and the promotion of pawns. Additionally, the board may be preset before the start of play, permitting the examination of "book" plays. To maximize execution speed, the program is written in assembly language (by SOFTWARE SPECIALISTS of California). Full graphics are employed in the TRS-80 version, and two widths of alphanumeric display are provided to accommodate North Star users.

STARTREK 3.2 (Available for all computers) — Price: $ 9.95 Cassette, $13.95 Diskette

This is the classic Startrek simulation, but with several new features. For example, the Klingons now shoot at the Enterprise without warning while also attacking starbases in other quadrants. The Klingons also attack with both light and heavy cruisers and move when shot at! The situation is hectic when the Enterprise is besieged by three heavy cruisers and a starbase S.O.S. is received! The Klingons get even!

SPACE TILT (Apple only) — Price: $10.95 Cassette, $14.95 Diskette

Use the game paddles to tilt the plane of the TV screen to "roll" a ball into a hole in the screen. Sound simple? Not when the hole gets smaller and smaller! A built-in timer allows you to measure your skill against others in this habit-forming action game.

GAMES PACK I and GAMES PACK II — Price: $ 9.95 each, Cassette; $13.95 each, Diskette

GAMES PACK I contains BLACKJACK, LUNAR LANDER, CRAPS, HORSERACE, SWITCH and more. GAMES PACK II includes CRAZY EIGHTS, JOTTO, ACEY-DUCEY, LIFE, WUMPUS and others. Available for all computers.

Why pay $5.95 or more per program when you can buy a DYNACOMP collection for just $9.95?

STUD POKER (ATARI only) — Price: $11.95 Cassette, $15.95 Diskette

This is the classic gambler's card game. The computer deals the cards one at a time and you (and the computer) bet on what you see. The computer does not cheat and usually bets the odds. However, it sometimes bluffs! Also included is a five card draw poker betting practice program. This package will run on a 16K ATARI.

STATISTICS and ENGINEERING

DATA SMOOTHER (Available for all computers) — Price: $14.95 Cassette, $15.95 Diskette

This special data smoothing program may be used to rapidly derive useful information from noisy business and engineering data which are equally spaced. The software features choice in degree and range of fit, as well as smoothed first and second derivative calculation. Also included is automatic plotting of the input data and smoothed results.

FOURIER ANALYZER (Available for all computers) — Price: $14.95 Cassette, $15.95 Diskette

Use this program to examine the frequency spectra of limited duration signals. The program features automatic scaling and plotting of the input data and results. Practical applications include the analysis of complicated patterns in such fields as electronics, communications and business.

TFA (Transfer Function Analyzer) — Price: $19.95 Cassette, $23.95 Diskette

This is a special software package which may be used to evaluate the transfer functions of systems such as hi-fi amplifiers and filters by examining their response to pulsed inputs. TFA is a major modification of FOURIER ANALYZER and contains an engineering-oriented decibel versus log-frequency plot as well as data editing features. Whereas FOURIER ANALYZER is designed for educational and scientific use, TFA is an engineering tool. Available for all computers.

FOURIER ANALYZER and TFA may be purchased together for a combined price of $29.95 (Cassettes) and $37.95 (Diskettes).

REGRESSION I (Available for all computers) — Price: $19.95 Cassette, $23.95 Diskette

REGRESSION I is a unique and exceptionally versatile one-dimensional least squares "polynomial" curve fitting program. Features include very high accuracy; an automatic degree determination option; an extensive internal library of fitting functions, data editing, automatic data and curve plotting, a statistical analysis (e.g., standard deviation, correlation coefficient, etc.) and much more. In addition, new fits may be tried without reentering the data. REGRESSION I is certainly the cornerstone program in any data analysis software library.

REGRESSION II (PARAFIT) (Available for all computers) — Price: $19.95 Cassette, $23.95 Diskette

PARAFIT is designed to handle those cases in which the parameters are embedded (possibly nonlinearly) in the fitting function. The user simply inserts the functional form, including the parameters (A(1), A(2), etc.) as one or more BASIC statement lines. Data and results may be manipulated and plotted as with REGRESSION I. Use REGRESSION I for polynomial fitting, and PARAFIT for those complicated functions.

REGRESSION I and II may be purchased together for $36.95 (cassettes) and $44.95 (diskettes).

Availability

DYNACOMP software is supplied with complete documentation containing clear explanations and examples. All programs will run within 16K program memory space (ATARI requires 24K). Except where noted, programs are available on ATARI, PET, TRS-80 (Level II) and Apple (Applesoft) cassette and diskette as well as North Star single density (double density compatible) diskette. Additionally, most programs can be obtained on standard 8" CP/M floppy disks for systems running under MBASIC.

BUSINESS and UTILITIES

MAIL LIST II (North Star only) — Price: $21.95

This many-featured program now includes full alphabetic and zip code sorting as well as file merging. Entries can be retrieved by user-defined code, client name or Zip Code. The printout format allows the use of standard size address labels. Each diskette can store more than 1100 entries (single density, over 2300 with double density systems)!

TEXT EDITOR I (Letter Writer) — Price: $14.95 Cassette, $18.95 Diskette

An easy to use, line-oriented text editor which provides variable line widths and simple paragraph indexing. This text editor is ideally suited for composing letters and is quite capable of handling much larger jobs. Available for all computers.

FINDIT (North Star only) — Price: $19.95

This is a three-in-one program which maintains information accessible by keywords of three types: Personal (e.g., last name), Commercial (eg: plumbers) and Reference (eg: magazine articles, record albums, etc). In addition to keyword searches, there are birthday, anniversary and appointment searches for the personal records and appointment searches for the commercial records. Reference records are accessed by a single keyword or by cross-referencing two or three keywords.

DFILE (North Star only) — Price: $19.95

This handy program allows North Star users to maintain a specialized data base of all files and programs in the stack of disks which invariably accumulates. DFILE is easy to set up and use. It will organize your disks to provide efficient locating of the desired file or program.

COMPARE (North Star only) — Price: $12.95

COMPARE is a single disk utility software package which compares two BASIC programs and displays the file sizes of the programs in bytes, the lengths in terms of the number of statement lines, and the line numbers at which various listed differences occur. COMPARE permits the user to examine versions of his software to verify which are the more current, and to clearly identify the changes made during development.

COMPRESS (North Star only) — Price: $12.95

COMPRESS is a single-disk utility program which removes all unnecessary spaces and (optionally) REMark statements from North Star BASIC programs. The source file is processed one line at a time, thus permitting very large programs to be compressed using only a small amount of computer memory. File compressions of 20-50% are commonly achieved.

GRAFIX (TRS-80 only) — Price: $12.95 Cassette, $16.95 Diskette

This unique program allows you to easily create graphics directly from the keyboard. You "draw" your figure using the program's extensive cursor controls. Once the figure is made, it is automatically appended to your BASIC program as a string variable. Draw a "happy face", call it H$ and then print it from your program using PRINT H$! This is a very easy way to create and save graphics.

TIDY (TRS-80 only) — Price: $10.95 Cassette, $14.95 Diskette

TIDY is an assembly language program which allows you to renumber the lines in your BASIC programs. TIDY also removes unnecessary spaces and REMark statements. The result is a compacted BASIC program which uses much less memory space and executes significantly faster. Once loaded, TIDY remains in memory; you may load any number of BASIC programs without having to reload TIDY!

SIMULATIONS and EDUCATION

BLACK HOLE (Apple only) — Price: $14.95 Cassette, $18.95 Diskette

This is an exciting graphical simulation of the problems involved in closely observing a black hole with a space probe. The object is to enter and maintain, for a prescribed time, an orbit close to a small black hole. This is to be achieved without coming so near the anomaly that the tidal stress destroys the probe. Control of the craft is realistically simulated using side jets for rotation and main thrusters for acceleration. This program employs Hi-Res graphics and is educational as well as challenging.

VALDEZ (Available for all computers) — Price: $14.95 Cassette, $18.95 Diskette

A simulation of supertanker navigation in the Prince William Sound and Valdez Narrows. The program uses an extensive 256X256 element radar map and employs physical models of ship response and tidal patterns. Chart your own course through ship and iceberg traffic. Any standard terminal may be used for display.

FLIGHT SIMULATOR (Available for all computers) — Price: $17.95 Cassette, $21.95 Diskette

A realistic and extensive mathematical simulation of take-off, flight and landing. The program utilizes aerodynamic equations and the characteristics of a real airfoil. You can practice instrument approaches and navigation using radials and compass headings. The more advanced flyer can also perform loops, half-rolls and similar aerobatic maneuvers.

TEACHER'S PET I (Available for all computers) — Price: $ 9.95 Cassette, $13.95 Diskette

This is the first of DYNACOMP's educational packages. Primarily intended for pre-school to grade 3, TEACHER'S PET provides the young student with counting practice, letter-word recognition and three levels of math skill exercises.

Ordering Information

All orders are processed and shipped postpaid within 48 hours. Please enclose payment with order along with computer information. If paying by VISA or Master Card, include all numbers on card. For orders outside North America add 10% for shipping and handling.

Add $2.50 to diskette price for 8" floppy disk (soft sectored, CP/M, Microsoft BASIC)

Deduct 10% when ordering 3 or more programs.

Ask for DYNACOMP programs at your local software dealer. Write for detailed descriptions of these and other programs from DYNACOMP.

DYNACOMP, Inc.
6 Rippingale Road
Pittsford, New York 14534
(716) 586-7579

VISA — master charge

New York State residents please add 7% NYS sales tax.

CIRCLE 136 ON READER SERVICE CARD

Casheab, continued...

The manual is over 70 pages long, and discusses how to install board jumpers to modify the options, how to use the synthesizer hardware, the software which is provided with the boards, theory of operation of the synthesizer (and some other theory too!), references, maintenance procedures, parts lists, and a listing of a sample test program. Schematics and parts layouts are supplied separately. The boards are somewhat densely populated, but the layout is clean, and there are no last minute changes to the PC layout strewn about, as is so common with initial production units.

The software is CP/M compatible and is mostly written in Microsoft Basic, consisting of three main parts: the waveform generator, the score generator, and the Play program. Source code is provided for all software.

The waveform generator uses a Fast Fourier Transform (FFT—see "Fast Fourier Transforms on Your Home Computer", W.D. Stanley and S.J. Peterson, *Byte*, December 1978) to transform user-specified harmonic intensities into a waveform suitable for loading into the synthesizer. Attack and decay envelopes can also be specified (64 values in the range 0-255 each for attack and decay), allowing a particular waveform to be customized into a complete timbre. During my preparation for this article, Casheab suggested that I try to generate waveforms by simply adding the weighted harmonic waveforms. Compiling turned out to take 7 seconds per harmonic (running under UCSD Pascal—more on that later), so that ten harmonics took about a minute, as opposed to about 5 minutes with the FFT. The FFT program running under interpreted Basic takes about 15 minutes to compute a waveform; under compiled Basic it takes about 5 minutes. Casheab may be supplying such a program with their synthesizer by the time this article is in print. Both harmonics and timbres may be saved on disk.

The score generator accepts score notation as character strings in Basic DATA statements, and produces a HEX file as output for the Play program. The notes are represented as SANXOTMS, where 'S' represents a possible slur; 'A' the amplitude of the note (0 [off] through 9); 'N' what note (A, B, C, D, E, F, G) is to be played; 'X' whether the note is sharp, flat or natural; '0' the octave number (0 through 6); 'T' the duration (time) of the note; and 'M' whether the note is dotted or not. Thus, a 'typical' note might be given as '3F#4Q.—', meaning that with amplitude 3, play an F# in the fourth octave as a dotted quarter note with a post-slur. The number of voices to be scored is specified, as is which channel is to be used by each voice and which voices are FM modulators. The 'notes' for each voice are then listed sequentially, with an 'X' to terminate each voice and an 'E' to terminate the piece. Some typographical errors are flagged by the program as errors.

The Play program is the only program written in 8080 assembly language, and it ties the timbres and the scores together. It allows a score to be read into memory and timbres associated with each channel. Channel assignments can be modified, as can FM modulation flags, and attack/decay envelopes can be scaled. The piece may be started and stopped, and when stopped, the amplitude of the piece and its tempo may be varied. This software works just fine for up to about 5 voices, but for more than that, it is recommended that a real-time clock be available to the Play program in order to produce timing which is truly even. I didn't have a real-time clock and didn't want to purchase one, so I rigged up a 555 timer chip as a variable—frequency square wave oscillator controlled by a potentiometer to provide synchronization to the software via an input port. Casheab supplies two versions of their software, one for use with systems with the 8253 real-time clock and one for systems without. Since source is supplied, you could modify the code for the 8253 to work with your own real-time clock.

The boards are somewhat densely populated, but the layout is clean, and there are no last minute changes to the PC layout strewn about.

The procedure for playing a piece is somewhat involved. You first create a series of Basic DATA statements, using a text editor, to represent the music you want to play. You then run the score program to create a score file. If you need new timbres for the piece, you run the waveform program to generate them. Finally, you run the Play program to hear your music. If an error is made in the score, you must start again at the editor, then the score program, and then the Play program. Despite some inconveniences, however, the software which is delivered with the synthesizer is sufficient to allow one to encode and play any piece of up to 32 simultaneous voices.

Debugging musical pieces in this fashion is very interesting, since the scores are quite like programs, and you must listen to your 'program' to discover the mistakes which you have made. A quarter note which was written as a eighth note will result in one voice 'sliding' earlier by a eighth of a beat for the remainder of the

song, usually causing some discord, and not revealing its exact location in an obvious fashion.

The syntheisizer comes complete with the above software and some musical pieces and timbres ready to play. A Bach Fugue and Prelude are included, as is "Pictures at an Exhibition" and the theme from Star Wars. Casheab also has coded a Bach Two-Part Invention, but it was not on my initial distribution disk. Timbres supplied included trumpet and clarinet, but it is relatively easy to construct new timbres from information in the literature (either *Computer Music Journal*, or text books on acoustics).

The synthesizer did not work at all when I first plugged it in, but a call to Casheab indicated that my processor (an Ithaca Audio Z-80 board) was one of those which required a modification to the clock phase and sense jumpers. After I removed the jumper from JP15 to JP17 and added a jumper from JP16 to JP17, it worked immeditely and correctly. In fact, one surprising thing about this product is that it does EXACTLY what its documentation says it will; not much more, but certainly nothing less. I am used to a certain amount of 'puffing' in my sales brochures, but Casheab delivered exactly what they said, no excuses about "we're still working on it" or some such.

Use of the FM feature probably needs some explanation. Since an FM channel modifies the rate at which the channel two above it is scanned out, an FM channel running at low frequencies can be used to create vibrato in the modulated channel. If one places a sine waveform in channel 0 running at a low rate, say 1 Hz, then the sound coming out of channel 2 will warble slightly as its frequency changes. In order to facilitate this, I modified the syntax of the score program to allow frequencies in the range of 0.3 Hz to about 12 Hz to be specified directly instead of by using the score program note notation.

A more interesting use of the FM facility is to do 'real' FM music synthesis with it, as described in "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation" by John M. Chowning (*Computer Music Journal*, Vol 1, No 2). This technique uses a frequency which is a fraction of the carrier frequency as a modulating signal; that is, if you want to hear a

Free
HEATHKIT
CATALOG

PLACE STAMP HERE
Post Office
will not deliver
mail without
postage

Heath Company
Benton Harbor, MI 49022

1000—Hz tone, you concurrently modulate it with a 500—or 250—Hz tone. This has the effect of 'spreading' the spectrum of the carrier (1000—Hz) tone so that it has many rich harmonics, even if the two waveforms being considered are simply sine waves! This means that one makes decisions about the characteristics of one's FM timbres by modifying the ratios of the carrier frequency to the modulating frequency, not by changing the harmonic content of either. Additional ways to modify the FM timbres include use of non-sine waveforms for the carrier (but not for the modulator!) and modifying the degree of FM modulation throughout the duration of the note by modifying the attack/decay envelope of the modulating tone.

The software which is delivered with the synthesizer is sufficient to allow one to encode and play any piece of up to 32 simultaneous voices.

Other effects which can be obtained include echo and chousing. Echo can be obtained simply by repeating the notes for a voice on a second channel at a lower amplitude and with a short delay (rest) inserted before the start of the second channel. This effect can be very pleasing with organ fugues, for instance.

Chorusing is an effect which makes you think that more than one instrument is playing a voice. One problem with a digital synthesizer is its precision: twenty identical voices played at once sounds just like one loud voice. In order for a chorusing effect to work, the voices must play at slightly different frequencies, and so I modified my score program to provide three equally tempered scales, each off from the next by about 2 Hz. This allows me to have up to three channels playing the same voice but with distinct frequencies. Adding a small amount of vibrato (FM) fo each channel at a different vibrato frequency allows a reasonable chorusing effect to be obtained. Unfortunately, three FM'ed channels requires six channels for a single voice, making use of these effects somewhat complex and inefficient.

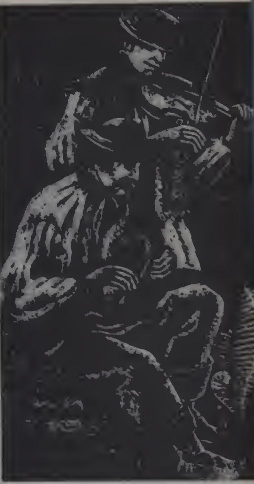
One final effect which I have not yet tried is to use a large number of channels, say eight, to control the harmonics of a single note individually. Using this technique, one can control the amplitudes of each harmonic of the note throughout the duration of the note, allowing very accurate synthesis of real musical instrument voices. Unfortunately, the Casheab could only support four voices which required control of eight harmonics each.

As stated above, I wrote the waveform program in UCSD Pascal, and in fact rewrote the entire software system in UCSD Pascal, combining it into a single program in order to be able to customize it more easily; only portions of the Play program had to remain in assembly language. The Casheab software takes advantage of single-character keystrokes for command selection, just like UCSD Pascal, but it does not take advantage of the random addressing capability of most CRT systems. My new synthesizer software does, and maintains tables of information about the synthesizer and score on the screen at all times. The cumbersome Basic DATA statement formats and line numbers were replaced with free-formats and no line numbers. Also added as a screen-oriented note editor which allows one to halt the score in the middle of play and see the notes which were then being played displayed on the screen. Those notes and notes near to them in time can then be modified and the score re-played, short-circuiting the laborious edit cycle described above. A channel-inhibit feature was also added in order to facilitate debugging multi-voiced pieces. Casheab owners who are interested in running this software can contact me at the address given at the beginning of the article.

One thing which modifying the software showed me was that the Casheab software does not BEGIN to take advantage of the flexibility which the Casheab hardware could provide. As more people purchase Casheab systems, software should develop to allow really innovative uses.

One obvious augmentation of the current Casheab system would be to allow a keyboard to be played 'through' it to simulate a sophisticated organ, or better. Casheab is aware of this, and has a general-purpose slave processor card (also S-100), which could be used as a smart keyboard-scanning card, implemented in wire-wrap form at the moment. It contains a down-loadable Z-80 system and 16K bytes of RAM, with I/O ports and some breadboarding space for placement of multiplexers and cable connectors. Software to run the synthesizer from the keyboard is working at this time, but no product using either this hardware or software has been announced yet.

The current score syntax does not allow for modifications to the tempo of the piece while it is being played; nor does it allow for 'blue' notes, that is notes which glide between normal equal tempered note frequencies. Also, the current implementation ties a hardware channel to a voice, a restriction which is really unnecessary. With the current software, it is not possible for a note to decay at the same time that another note for the same voice is attacking; that would require one channel to be playing two notes at once. Software to provide dynamic channel allocation would allow this kind of attack/decay overlap.



In addition, the software is written to 'simulate' organ notes rather than percussive notes such as harpsichord or piano. With an organ, the note starts to attack when you hit the key, rises to a sustain level, and stays at that sustain level until the note is released, at which point it decays. With a piano note, the note attacks and then decays for the note duration, a completely different effect. Changes to the attack/decay software will be necessary fully to support percussive instruments.

There is no reason why the Casheab hardware cannot support any of these new concepts, or even more, but the software is not yet available to support them.

All in all, I would say that the Casheab is a high-quality piece of hardware, well thought out, well designed, and well implemented. The software is complete but somewhat Spartan, demonstrating the capabilities of the Casheab hardware, but really serving to provide a starting point from which serious computer musicians can depart. It is a unique and reasonably priced S-100 bus board which all computer musicians with S-100 bus systems should investigate. (Casheab, 5737 Avenida Sanchex, San Diego, CA 92124, (714) 277-2547.) □

Another Look at Educational Software and Books

David Lubar

Books are easy to deal with; they've been around a long time, and you pretty much know what to expect from them. Software is another matter. The computer can be used merely to replace a book, or to take some of the load off of a teacher. Is this approach desirable? This question, and other related problems, will have to be answered sooner or later by both the software writers and the reviewers. With this unanswered problem peering over my shoulder, it's time to take a look at some educational material.

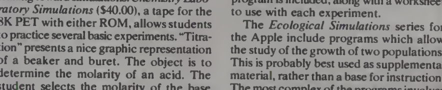
Texas Instruments, P.O. Box 10508, Lubock TX 79408, seems to be heading in the right direction with their *Early Reading* cartridge ($54.95). This plug-in module, which requires the speech synthesizer, combines excellent speech capability with good programming. The child using the program has three options. First option: He can select a picture, then hear a story about the picture. Before each story, he is taught several words. The words are shown on the screen and spoken by the computer. Then the child must pick the correct word from a sentence. Once this is done, he sees and hears the story. The different parts are accompanied with graphics and animation. Finally, the story is shown again without any speech, giving the child a chance to read on his own. Second option: sentences are presented with missing words. The child must pick the correct answer from three choices. The third option presents a story. At certain parts, the child is given a choice of how to end a sentence. His selection determines the direction the story will take. At the end, questions are presented. If the child has trouble, the program branches back to the portion of the story which covers the answer. This package seems aimed at children who have some reading skills. With the aid of a parent, however, even children who haven't yet learned to read could enjoy the program.

Conduit, P.O. Box 388, Iowa City, Iowa 52244 distributes educational software for the college level. Their full line is too wide to cover here, but a representative sample can be mentioned. In general, they combine instruction and experimentation with a leaning toward simulations. *Chemistry Laboratory Simulations* ($40.00), a tape for the 8K PET with either ROM, allows students to practice several basic experiments. "Titration" presents a nice graphic representation of a beaker and buret. The object is to determine the molarity of an acid. The student selects the molarity of the base and the ph at which the indicator changes. Inappropriate responses result in short lessons. The tape also contains three other programs which cover reduction-oxidation, kinetics, and the Bohr atom.

The manual that comes with the tape is well done. It describes the program and steps through a sample run. A listing of the program is included, along with a worksheet to use with each experiment.

The *Ecological Simulations* series for the Apple include programs which allow the study of the growth of two populations. This is probably best used as supplemental material, rather than a base for instruction. The most complex of the programs involves the interaction of several plants and animals on a tundra. Conduit gets high points for

A scene from Titration.



ease of use and interaction. Each screen has a menu at the bottom. The student can move from simulation to discussion, then return to the simulation and change various parameters. A scrolling line graph shows the growth of each organism. Again, the documentation is very good.

Data Command, 670 W. Broadway, Bradley IL 60915, has taken the approach of combining games and instruction. The general technique is to explain the game and give some background on the area being covered. After this, the program takes the form of a quiz where each correct answer results in a score for the student in the game. *Tennis Anyone?* has units that cover plurals, prefixes, suffixes, contractions, homonyms, and base words. After each question, a graphic player volleys with a graphic paddle that represents the student. If the answer is correct, the student wins the point; if he is wrong, he loses. This isn't a bad approach, but it would be nice if the student could take more part in the actual game. He has no control over his half of the court. Still, this combination of game and test will probaly hold a user's interest for longer than straight testing. The programs are available on disk for a 32K Apple with ROM Applesoft and disk and tape for the TRS-80 (32K required for disk, 16K for tape). The units are available individually for $29.95 each, or in a set of six, covering all the subjects mentioned above, for $170.75.

Tycom Associates, 68 Velma Ave., Pittsfield MA 01201, produces software for the 8K PET. Their programs are straight drill and practice. *French* ($15.95), is well done. It presents several options. The user can see a French word for which he has to provide the English equivalent. Or he can see an English word, or the words can be mixed. Another mode presents multiple choice questions. At the end of a segment, the student is shown his percentage of correct answers. *Vocab* ($15.95) contains two programs that present words and ask for synonyms. A nice touch is the routine which checks for spelling mistakes. If the user gives an answer that is close to the correct one, he is told that he may have made a spelling mistake and is given another chance. On wrong guesses, he is given a clue. *Algebra* ($19.95) gives instruction in seven areas, including factoring and sets, and gives drills in each area. Unfortunately, the program crashes if you hit ENTER without first hitting a number. This sort of thing is easy to prevent and should not occur in professional software.

Compak, Inc, P.O. Box 14852, Austin TX 78761, has a math package for a 32K Apple with Applesoft and TI 99/4 computers. Basically, the program presents math problems, keeps records of student's scores, and provides help if the student has trouble. Students who do well are advanced to more difficult modules. The numbers are presented in large size on the screen. The concepts covered include addition of whole numbers, common fractions, and elementary algebra. The advanced areas contain some tutorial material, but the major function of the programs is drill and practice. On entry, the user is asked several questions, such as whether sound is desired or whether a printer is attached. The user is then asked to enter the level at which he wants to work. A number out of the range of the program will cause a crash. Answers to the math problems are entered from left to right, which seems somewhat unnatural when you are used to pencil and paper. The disks can be purchased in three fashions. The entire set, in either a concept-by-grade or grade-by-concept format costs $495.00. Disks with all concepts for a single grade cost $65.00, disks with one concept for all grade levels are $50.00.

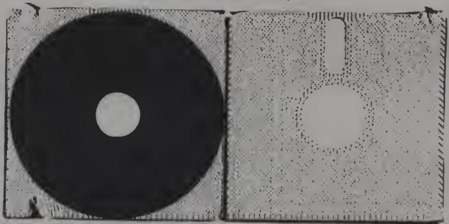
J & S Software, 140 Reid Ave, Port Washington NY 11050, is a producer of software for the high school and junior college level. The programs are a nice blend of question and instruction, with a well-designed branching feature. Whenever a student answers a question incorrectly, he is switched to a second program which presents more background for the questions. I only have two major criticisms of the programs. First, they won't accept "T" and "F" for "True" and "False." A minor problem, but annoying to anyone who is used to this type of entry. Also, when the student selects a unit, the computer will accept RETURN by itself as a response. Later, when it uses the input variable from this response, the null character will cause an error. Aside from these minor problems, the programs are nice. The material goes into a fair amount of depth. There are fifteen programs in both the Chemistry and Biology series. Individual programs in Chemistry cost $19.50, a set of six is $75.00, all in a set cost $150. The Biology units are $19.95 each, six for $80.00, and $160 for the whole set. The programs are available on disk or tape for a 32K Apple with Applesoft.

Pulpware

And then there are the books. An interesting and varied assortment of educational material is on hand. *Computeronics* is a series of workbooks and manuals that deal with computers. They are well done and very reasonably priced. Some of the books cover general computer literacy, others go into a fair amount of programming instruction, showing how to work with Basic. My favorite, a book called *Sidetrips*, presents twenty-five mind teasers, along with solutions in Basic, followed by sections that cover text-type graphics and debugging. A final section gives programs and asks the student to figure out what they do. Quite a bargain for $2.85. The highest priced item is one of the teacher's manuals. At $10.10, even this is reasonable, especially when compared to the average price of textbooks. For more information, contact **Gifted Child Project**, c/o Seminole Blueprinting, 1212 North Monroe St., Tallahassee FL 32303.

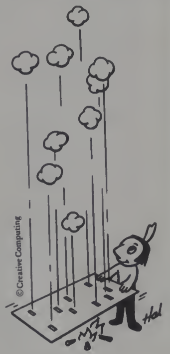
Computer Parts Kit from **The Educational Computer Shoppe**, Route 3, Box 601, Cambridge MN 55008, contains a slim activities book and a lot of fantastic stuff to examine and enjoy. The $38.50 kit includes parts from old computers, hard and floppy disks which can be examined and dissected, paper tape, and various examples of vendor literature. Craig Solomonson, who created the kit, came up with a real stroke of genius in one of the activities. A bottle of developer is included which can be applied to tape, thus showing the pattern of stored information. Once the pattern is developed, it can be lifted off with Scotch tape and saved. A teacher with a bit of background in the field could get a lot of mileage from this kit.

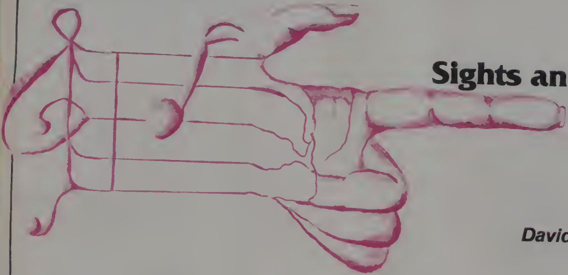
A dissected disk from the computer parts kit.



Teaching Basic Bit by Bit from the **Lawrence Hall of Science**, Math and Computer Education Project, University of California, Berkeley CA 94720, contains lessons and activities to help teachers present a course on Basic. While the examples are designed for Apple and PET computers, the book can be used for any system. Each section starts with a list of the activities and new concepts that will be covered, as well as prerequisites for and goals of the section. Whether it is used as the basis for a course, or just for background ideas, the book is well worth the price of $7.50 (plus $2.00 p&h per order).

Resource Software International, Inc, 140 Sylvan Ave., Englewood Cliffs NJ 07632, markets software in the form of books. They presently have 24 packages designed for use with **CP/M** (a registered trademark of Digital Research) systems with Microsoft Basic, and are planning versions for home computers such as the Apple. Each pamphlet contains a flowchart, program listing, and teacher's guide. *Learning and Practicing with Fractions* gives word problems with multiple-choice answers. Again, we are dealing to a large extent with drill and practice. On the nice side, the program presents a pretest, and also shows the student which problems he should practice. The program listings seem to be Xeroxed, and on some of the pages, the top line was cut off. Considering the availability of ready-to-run software, this approach does not seem to be the best way to market programs in the educational area. □





Sights and Sounds at PCAF

David Lubar

Above the crowd that surged through the depths of the Philadelphia Civic Center, in an inconspicuous auditorium on the first floor, great things were happening. Through hard work, and with a lot of volunteers, the Personal Computer Arts Festival turned out to be a worthwhile experience. The Festival, run by Steve Levine, Dick Moberg, and Bill Mauchly, went for two full days; Saturday was dedicated to music, Sunday to graphics. The show opened with Frank Covitz and Cliff Ashcraft, creators of the MTU music synthesis software. They demonstrated some new features of their software, using a Kim and Aim. To show the versatility of the system, they played several compositions, including Handel's *Water Music* played with an instrument that sounded like a trumpet being blown under water (wumpet?), and a classical piece done with an excellent synthesis of a harpsichord. In anticipation of the inevitable question, "Can your system only play classical music?", they also performed *Root Beer Rag* and *Foggy Mountain Breakdown*.

Next came Mike Riley from A B Computers, who demonstrated the Visible Music Monitor for the PET. This is a nice piece of software which aids in entering music for four-voice systems. It displays the notes and allows the user to define keys on the keyboard as notes to further simplify entry.

Many of you are probably familiar with the work of the next speaker, Hal Chamberlin. He's been involved in computer music for quite a while, and his newest venture is astounding. Hal is working on non-realtime music synthesis, or as he prefers to call it, computed music. The music is compiled over a long period of time; an hour or more might be required for one song. The results are saved on disks, many disks. With a sampling rate of 20 hertz, each 8 inch disk can hold 20 seconds of music. Using two drives, and inserting disks at a rapid pace, Hal gave the audience a sample of the music the system is capable of producing. It sounded great. While this might seem like a lot of trouble to go through, any system that can produce these results, and is capable of 132 voices, is worth the effort.

The high resolution display was reminiscent of the cartoons they used to make in the 40's and 50's.

Hal also provided some interesting insights in the field of computer music. At one time, there was a gap between the amateur and the pro. But with the advent of computed music, this gap is closing. Amateurs will soon be able to afford systems that sound as good as mega-buck synthesizers.

Many people have been looking for a music system for the TRS-80 that is comparable to available 6502 systems. Stewart Newfeld from Newtech demonstrated music systems for TRS-80 and S-100 systems. He explained that since these computers usually don't have high-resolution graphics capabilities, the entry system uses letters and numbers. The entry is done vertically, scrolling down as notes for the four voices are entered. A group of nice programming touches ease the pain of entry somewhat. The Newtech Music Box for the TRS-80 will be reviewed here in the near future.

A valuable contribution to the festival, and the programming world in general, was made by Rebecca Mercuri of RCA, who discussed the various editors available on music systems. She feels that, at the moment, they are all lacking to some degree. Rather than just point out flaws, she came with an example; the Bach (Basic algorithms for composing harmonics) system written by Michael Keith. This displays an entire page of music in black on white, and ideally, could be used with any music board. Their demonstration was on an Apple using ALF boards. Since Ms. Mercuri has an in-depth article on music editors in these very pages, only a brief mention of her requirements for a music editor will be made here. Among other requirements, a good music editor must have audio feedback, a simple file system for blocks of music, and full screen-editing. And the system should not be designed in such a way that one careless keystroke can wipe out a whole file.

Moving into the realm of advanced synthesis, the audience was treated to a demonstration of the Fairlight CMI, courtesy of Steve Levine and Bill Mauchly. This eight-voice synthesizer has some amazing abilities. A waveform can be drawn on the CRT with a light pen and then be played. Up to 64 harmonics can be specified for any envelope, and natural sounds can be digitized for recreation or modification. The system consists of a dual-level organ keyboard, computer terminal, and dual disk drives. The price (over 30K) puts it out of range of us mortals, but serious musicians and those involved in studio work will find the Fairlight to be worth a close look.

Dropping nearer to the price range of us mortals, John Bondy demonstrated the

Two Great New TRS-80 Computers. Radio Shack—Setting New Standards For Price and Performance!

13-Inch Color TV Receiver Available Separately. Only $399*

16K TRS-80 Model III. The new standard in personal computers—packed with the advanced features you want, yet priced unbelievably low! The Model III's handsome *one-piece* cabinet includes a high resolution 12-inch monitor that displays 16 lines of 64 upper/lower case characters, and graphics. It also includes a 65-key typewriter-style keyboard with data pad. There is room inside for two optional double-density disk drives—and you can add two more external drives for a total of approximately 670K bytes of on-line storage. Model III comes with a two-speed cassette interface, so most of our huge library of Model I, Level II software can be used. And the new 1500-baud mode gives you really fast cassette I/O. The very powerful Model III BASIC language, contained in 14K of ROM, features automatic line numbering, tracing, PEEK and POKE, machine language programming, keyboard-controlled screen print, I/O device routing, repeating keys, special characters and lots more. Other Model III features include a built-in real time clock, cassette jack and Model III bus, plus a built-in printer interface for direct connection with Radio Shack printers. With the built-in RS-232 firmware and the optional internal RS-232 board, communications is a snap! Model III is expandable to 48K RAM at any time, and is also available with 4K RAM and Level I BASIC (for only $699*). A full-blown 32K, 2 disk system with RS-232 is yours for only $2495.* This is the breakthrough you've been waiting for. Come in and try Model III today. **16K TRS-80 Model III . . . . .Only $999***

16K TRS-80™ Color Computer. Vivid 8-color graphics, sound, the high performance 6809E microprocessor and more—at a very affordable low price! This amazingly portable new micro attaches in seconds to any TV set. Use it at work, at school or at home. Its Extended Color BASIC language provides unique commands for graphics manipulations, easy machine language programming, complex musical sounds (with more than one note!), PEEK and POKE, specific error messages and much more. A built-in RS-232 serial interface lets you attach a modem for communications or a serial printer. Of course, you can save your programs on cassette (at a fast 1500 baud)—or just pop in one of our instant-loading ROM Program-Paks for a wide variety of fast-paced games and valuable personal applications. Like all TRS-80s, it's easy to expand—add a printer now, disk drives are on the way. An optional pair of joysticks brings extra maneuverability and real excitement to computer games. The TRS-80 Color Computer is also available with 4K RAM and Color BASIC (at just $399*). This is the multipurpose computer you've been waiting for. Try it today at your nearest Radio Shack store, Authorized Dealer or Computer Center. **16K TRS-80 Color Computer with Extended Color BASIC . . . . .Only $599***

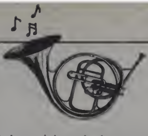
Radio Shack®

The biggest name in little computers™

A DIVISION OF TANDY CORPORATION

CIRCLE 146 ON READER SERVICE CARD

*Retail prices may vary at individual stores and dealers. Ask for our FREE 1981 TRS-80 Computer Catalog or Write Dept. 81, 1300 One Tandy Center, Fort Worth, Texas 76102



Sights & Sounds, continued...

Casheab Digital Music Synthesizer, which is designed for the S-100. Selling for $900 at the show (slightly higher elsewhere), the Casheab runs on Microsoft Basic and 8080 Assembly Language under CP/M. A Pascal version has also been produced. Notes are entered using alpha-numeric notation. The sound quality is decent, and the degree of user control fairly high, but to me the Casheab seems overpriced for what it delivers.

Then came demonstrations from Mountain Hardware and Syntauri Ltd. Their systems were covered last month so I won't go into detail here. The next speaker was Laurie Spiegel, a Juilliard graduate who has been involved with electronic music for about ten years. She spoke of the problems of present systems, including the limitation of standard notation to discreet steps. Demonstrating her own system, using an Apple for control and AlphaSyntauri keyboard for input, she produced some interesting and unusual effects. Ms. Spiegel is also interested in another type of synthesis, one between manufacturers of different music systems. In a conversation, she suggested that the creators of various systems should get together, combining their strong points. For example, it would be worth combining the MTU software with it's great waveform capabilities and the Mountain Hardware Board with it's great sound potential, forming a system that makes full use of the best side of both items. Add to this a non-discreet editor, and a keyboard such as the AlphaSyntauri, and you'll be heading in the direction of the sort of system Ms. Spiegel talked about.

Closing out the day, Kevin Doren showed the Crumar/MT1, another synthesizer aimed at a professional market, with 32 oscillators and FM synthesis capability. It was a nice, flexible system, that produced great sounds, but it's another member of the pro league that we poor amateurs can only admire from afar.

Sunday was devoted to graphics. Arch Robison opened the day with a talk on perspective graphics, demonstrating ways of creating illusions of depth. He also showed a system for storing the relative locations of any point in a 3-D drawing and determining whether any part of a line segment was on the screen.

Eric Podietz from Digital Mercury showed a beautiful real-time system for generating video art. Patterns could be drawn, then moved on the screen with a joystick, leaving trails of splendid designs as the patterns moved and rotated. Drawing to music, Mr. Podietz held the audience in rapt attention as he created art on the Advent screen.

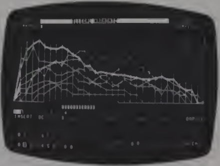
The art of 3-D sculpture was explained by David Dameron. Using a six-slot S-100 system and a homebuilt carver, he turns blocks of wax into busts and abstracts. The carver consists of a turntable, cutters, and a stepper motor capable of steps of one mil. This machinery costs around $900. Once a design has been entered into memory, the ratios can be changed, producing an elongated or truncated version of the sculpture. Mr. Damerson also showed some graphics done with a plotter. Not satisfied with traditional

Each 8 inch disk can hold 20 seconds of music.

methods, he made innovative use of this device. Instead of moving a pen, the plotter moves a stylus across a coated piece of metal. When the metal is bathed in acid, the result is a computer-created etching.

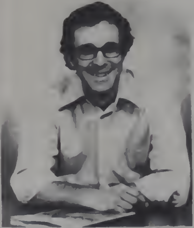
The next treat was a movie of a flight-simulator that runs on a GE graphics system. The high resolution display was reminiscent of the cartoons they used to make in the 40's and 50's. The pilot gets a real-time view from the cockpit, and the system could also show a side view of the jet. A second movie covered the MIT Genesis system. This was one of the early animation systems, producing results similar to what can be done today on an Apple or Atari. It was interesting to watch the operator create shapes, then program the system to animate these shapes.

There was more to come, but duty called and I had to hustle down to the Creative Computing booth. Those who are interested in participating in the Personal Computer Arts Festival or in hearing of future events can write to: Computer Arts Forum, C/O PACS, Box 1954, Philadelphia PA 19105. ☐



A display of wave form harmonics on the Fairlight.

A REMARKABLE MAGAZINE



David Ahl, Founder and Publisher of Creative Computing

creative computing

"The beat covered by Creative Computing is one of the most important, explosive and fast-changing."—Alvin Toffler

You might think the term "creative computing" is a contradiction. How can something as precise and logical as electronic computing possibly be creative? We think it can be. Consider the way computers are being used to create special effects in movies—image generation, coloring and computer-driven cameras and props. Or an electronic "sketchpad" for your home computer that adds animation, coloring and shading at your direction. How about a computer simulation of an invasion of killer bees with you trying to find a way of keeping them under control?

Beyond Our Dreams

Computers are not creative per se. But the way in which they are used can be highly creative and imaginative. Five years ago when *Creative Computing* magazine first billed itself as "The number 1 magazine of computer applications and software," we had no idea how far that idea would take us. Today, these applications are becoming so broad, so all-encompassing that the computer field will soon include virtually everything!

In light of this generality, we take "application" to mean whatever can be done with computers, *ought* to be done with computers or *might* be done with computers. That is the meat of *Creative Computing*.

Alvin Toffler, author of *Future Shock* and *The Third Wave* says, "I read *Creative Computing* not only for information about how to make the most of my own equipment but to keep an eye on how the whole field is emerging.

Creative Computing, the company as well as the magazine, is uniquely light-hearted but also seriously interested in all aspects of computing. Ours is the magazine of software, graphics, games and simulations for beginners and relaxing professionals. We try to present the new and important ideas of the field in a way that a 14-year old or a Cobol programmer can understand them. Things like text editing, social simulations, control of household devices, animation and graphics, and communications networks.

Understandable Yet Challenging

As the premier magazine for beginners, it is our solemn responsibility to make what we publish comprehensible to the newcomer. That does not mean easy; our readers like to be challenged. It means providing the reader who has no preparation with every possible means to seize the subject matter and make it his own.

However, we don't want the experts in our audience to be bored. So we try to publish articles of interest to beginners and experts at the same time. Ideally, we would like every piece to have instructional or informative content—and some depth—even when communicated humorously or playfully. Thus, our favorite kind of piece is acessible to the beginner, theoretically non-trivial, interesting on more than one level, and perhaps even humorous.

David Gerrold of *Star Trek* fame says, "*Creative Computing* with its unpretentious, down-to-earth lucidity encourages the computer user to have fun. *Creative Computing* makes it possible for me to learn basic programming skills and use the computer better than any other source.

Hard-hitting Evaluations

At *Creative Computing* we obtain new computer systems, peripherals, and software as soon as they are announced. We put them through their paces in our Software Development Center and also in the environment for which they are intended—home, business, laboratory, or school.

Our evaluations are unbiased and accurate. We compared word processing printers and found two losers among highly promoted makes. Conversely, we found one computer had far more than its advertised capability. Of 16 educational packages, only seven offered solid learning value.

When we say unbiased reviews we mean it. More than once, our honesty has cost us an advertiser—temporarily. But we feel that our first obligation is to our readers and that editorial excellence and integrity are our highest goals.

Karl Zinn at the University of Michigan feels we are meeting these goals when he writes, "*Creative Computing* consistently provides value in articles, product reviews and systems comparisons . . . in a magazine that is fun to read."

Order Today

To order your subscription to *Creative Computing*, send \$20 for one year (12 issues), \$37 for two years (24 issues) or \$53 for three years (36 issues). If you prefer, call our toll-free number, **800-631-8112** (in NJ 201-540-0445) to put your subscription on your MasterCard, Visa or American Express card. Canadian and other foreign surface subscriptions are \$29 per year, and must be prepaid. We guarantee that you will be completely satisfied or we will refund the entire amount of your subscription.

Join over 80,000 subscribers like Ann Lewin, Director of the Capital Children's Museum who says, "I am very much impressed with *Creative Computing*. It is helping to demystify the computer. Its articles are helpful, humorous and humane. The world needs *Creative Computng*."

creative computing

Attn: Cindy
P.O. Box 789-M
Morristown, NJ 07960
Toll-free **800-631-8112**
(In NJ 201-540-0445)

Siggraph

Bob Wallace

Both of us were excited to hear that this year's Siggraph conference would be near enough to visit, so we went down for a day to see what's new in computer graphics. The annual conference brings together a fascinating mixture of computer graphics enthusiasts, from personal computer users to university researchers to big computer marketing types. The most interesting things I saw, as a newcomer to Siggraph, were several new personal graphics computers and the computer film show.

Siggraph is the Association for Computing Machinery (ACM) Special Interest Group (SIG) on computer graphics. Like most of the ACM SIG's, this group publishes a newsletter and sponsors an annual conference. The name ACM always brings to my mind a group of computing machines, sort of like a labor union for robots, but it's really for people who work with computing machines, especially university people who can understand computer science papers. Siggraph was larger and more commercially oriented than I expected; it was refreshing to see an ACM event with new, real systems on display and with commercial as well as university research papers. Over 2,000 people came to see 98 exhibits, 17 technical sessions and 8 tutorials.

Many people were impressed to see IBM exhibiting a business color graphics system, feeling this area is now "blessed" and others will surely follow now that the field has been made "respectable." Digital Equipment Corporation had a new color graphics system, too. I was impressed by Hewlett-Packard's new graphics machine, with a high resolution display allowing over 4,000 colors. However, at $40,000, it's out of my range. Much of the show repeated this story; beautiful dense color graphics with interactive three-dimensional control at prices I could never afford. But there were exceptions.

Apple exhibited their graphics tablet, showing it being used to enter pictures on the new Apple III. Having an Apple II myself, I was happy to see them involved. Intelligent Systems Corporation displayed their Compucolor line. I wonder why the Compucolor has not been more successful in the personal computer market; it outperforms the Apple in many respects, and if more software and hardware had been available for it I would have chosen it instead.

For those with bigger (but still limited) budgets whose primary interest is graphics, Cromemco's new graphics card looks like the way to go. Cromemco pioneered inexpensive color graphics with the Dazzler, back in ancient times (about 1976). Their second generation system provides various display formats, including single-color resolution of 756 by 484 points (horizontal and vertical), and

Bob Wallace, Microsoft, 10800 Northeast 8th, Suite 507, Bellevue, WA 98004

sixteen-color resolution of 378 by 242. The sixteen colors can be chosen from a palette of 4,096. I remember trying to show the Dazzler output on our local cable TV channel, and discovering we couldn't directly connect the Dazzler's video output to the cable company's equipment. However, the new Cromemco Color Graphics Interface provides both studio quality interface signals and the ability to "lock in" to an external source of synchronization, overcoming this problem. However, the output will not directly drive a color TV, even with an RF modulator, since it is RGB (red-green-blue) instead of normal video. The company does sell 13 and 19 inch RGB monitors, however.

PERQ's detailed resolution is good enough for interlocking rosettes.

The Dazzler lacked interface software, too; getting it to display a line strained the abilities of the most dedicated bit-twiddler, because figuring out where to stick a byte in memory to turn on a point in the display requires a complex mapping algorithm. The Apple requires an equally complex algorithm, but I don't need to use it since I can use PLOT and other graphics commands from Basic. The new Cromemco system provides subroutine to do things like display polygons, circles, and other shapes, and shift, scale, and rotate them.

One of the most interesting booths was a joint production by Bally, Dave Nutting Associates, and Tom DeFanti of the University of Illinois at Chicago Circle. Bally, you may recall, started to play the "home computer" game with the Arcade, which would be expandable to become a fairly powerful system; however, only the initial video-game-plus-tiny-basic version was produced. Tom DeFanti has been in the computer graphics field for years, and helped produce some of the effects for Star Wars (I believe the trench sequence was done with his system). He designed a graphics-oriented language called Grass (Graphics Symbiosis System), which has been further refined as a suitable language for people (especially artists and educators) to create programs that generate and animate graphics images.

Grass approaches Basic more than any other language, but the kinship is not close; much of the power of Grass comes from the use of macros (a macro is a sort of textual subroutine). Since all commands are macros, adding new commands is easy. Also, programs look like character strings to the user; using strings as the fundamental building block gives the system an overall unity, as well as the ability to put together a string in a program and then execute it as another program. Creating programs and data out of the same unit (as is done in Lisp and Smalltalk) opens up many doors.

In Grass, graphic images live in little boxes; one can easily define a circle, polygon, etc., in a box. (One problem I noticed was when moving around patterns, their little boxes would move around with them, which may wipe out some other pattern on the screen.) The system allows several programs to run at once (multitasking), all of them changing the display, which allows creating a scene in which several patterns are moving and interacting. The system is pretty fast (for a microprocessor). Although it appears to the user that programs consist of many levels of macro expansion, there is a lot of compilation and floating-point hardware running invisibly underneath that speed up the process.

David Nutting Associates developed the UV-1 Zgrass computer using chips developed for the Bally Arcade and incorporating DeFanti's Grass language, now called Zgrass. The computer includes a Z-80, 48K of RAM, 16K of ROM, floating point processor, color monitor, music synthesizer, four (count 'em) joysticks, serial port, and assorted switches and lights. Color resolution is 320 by 204, using 4 out of 256 colors. Cost is somewhere around $3000. It's pretty new, so don't expect it in your neighborhood computer store soon. Dave Nutting Associates is at 527 West Golf Road, Arlington Heights, Illinois 60005.

My favorite computer at the show, however, was the PERQ, by Three Rivers Computer Corporation. It looks at first like an expensive (but not out of sight) black-and-white graphics machine. But the resolution is enormous (something like 500 by 1,000), there is a text editor for a whole typed page with multiple fonts at once, it is very fast, and includes a Winchester-type hard disk. It is the most powerful system I've seen that's designed and sold as a personal computer. Actually, the selling part is still to come, as Three Rivers is just getting into production (as of July), but contrary to industry rumors they are definitely alive and on the move.

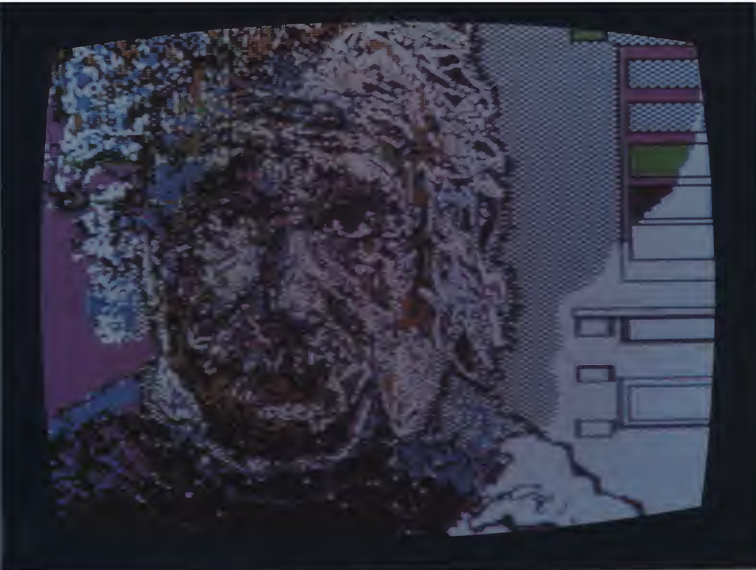
The PERQ processor directly executes Pascal P-code (at one per microsecond), which I found particularly appealing (since I am working on Microsoft's Pascal). Three Rivers Pascal is very powerful, having many added features like the ability to pass any length array as a parameter, as well as required extensions like strings and the ability to link together separately compiled routines. Pascal (like all current computer languages) is just too difficult for most people, but is perhaps the best language around for implementing large, powerful, interactive systems that lots of people will use.

Before leaving the exhibit area, we should mention a few more companies which make desktop graphics computers. Tektronix, because they have for years provided the graphics terminals many of us use. Chromatics, because their color terminals and systems are very nice if somewhat expensive. And Terak, for having an easy to use high resolution system based on the LSI-11.

PERQ computer (right) offers keyboard, super screen, winchester and lightning Pascal for about $20,000.

We attended one seminar, "trends in high performance graphic systems". It turned out to be on VLSI (very large scale integration, hundreds of thousands of gates on a chip), and its effects on future graphics machines. One theme was pairing each memory chip with a graphics processor chip, putting a large array of these pairs together executing in parallel. Lots of multiplying and dividing must be done to change a displayed image (move it around, rotate it, scale it, etc.) especially when such niceties as curved surfaces and shading must be handled as well, and these take time. Doing them all at once speeds up the process considerably. The last paper described how university students can design their own chips and have them produced and returned without waiting a year or two.

I wanted to attend the tutorials on low cost computer graphics and user interfaces to graphics systems. The latter topic is key, since as usual, the software in this field lags several years behind the hardware. In what ways does an artist or animator organize the visual field? What's the easiest way to input a design or shape? How do factors like color and time enter in? How can images be edited and combined to form



new ones? These questions need a lot of experimentation and creativity before those who work with pictures can use computers effectively.

The film show was the high point of the conference for me. Here the newest and most intricate applications of the computer artist's craft were shown to an enthusiastic audience. The group was quick to applaud sequences with successful execution of difficult graphics tasks; smooth curved color surfaces were always well received, and the few with shadows or highlights, moving in space, practically got a standing ovation. It made no difference

My favorite computer at the show was the PERQ, the most powerful system I've seen that's designed and sold as a personal computer.

whether the subject was an aviation simulation, a molecule, or a commercial for a car; if the graphics were good, the oohs and ahs kept coming.

The granddaddy of computer graphics companies, Evans and Sutherland, had a film demonstrating the state-of-the-art in interactive graphics. This means real-time, color, and three-dimensional visions of (for example) complex molecules. Three-dimensional in this case includes displaying two slightly different images, one for each eye, which (with the right viewer) look like one object floating in space. Looking "inside" such an object can be done by slicing off a part in front of the viewer (using the "hither" plane), and slicing off a part at the rear (using the "yon" plane), giving an arbitrary cross section of some depth. This is handy when studying a molecule or other object with internal structure. The amount of raw computing needed to do this in real time boggles the mind.

Activity around the Zgrass stand.

Rapt attention was generally reserved for those entries which were computer art for art's sake. I particularly enjoyed "Spiral", by Tom DeFanti; it reminded me of earlier works by Whitney, but taken further technically and with great control over timing and development. Several other "serious" art works were shown. Unfortunately the programs printed for the film series were lost, and I cannot remember the names of a few others that deserve mention.

More in the entertainment field, we'll

The Seventh Annual Conference on Computer Graphics and Interactive Techniques July 14-18, 1980, Seattle, Washington

Andrea Lewis

To the computer graphics inductee, the exhibits at Siggraph '80 are at first impressive, but fragmented. Image after image swirls by — auto body designs, revolving planets, graphs of functions, cartoon characters, world maps — on forbidding, "industrial looking" hardware. The imagination zooms as one wonders just how far this sophisticated technology can go. Is the day nearing when so much of man's intelligence has been deposited in micro-magnetic form to be played back in your choice of 4,095 colors and viewed

It's easy for a writer who still can't get over the marvel of word processing to extrapolate the advantages now available to designers, planners, and analysts in any field.

from *any* angle, that we really won't have anything left to do with ourselves except sit around in our sense-o-rama capsules? Well, maybe not, but it's easy for a writer who still can't get over the marvel of word processing to extrapolate the advantages now available to designers, planners, and analysts in any field. The ability to see all types of models, move them around, do projections, simulations and "what ifs" is sure to be a major factor in the ever-increasing rate of change in our hi-tech society.

Andrea Lewis, Microsoft, 10800 Northeast 8th, Suite 507, Bellevue, WA 98004

Judging from the exhibits at Siggraph '80, the range of what's available is very wide. On the high end, there are companies like Tektronix and Ramtek with high-resolution, plug-compatible intelligent terminals. An impressive high end terminal from Vector General, the 3033, mesmerizes the viewer with fast moving 3D color images of incredible resolution.

On the other end, you could see affordable graphics interface and display memory cards from Cromemco and Bally's Zgrass Graphics System. Zgrass is a user-oriented language especially designed for graphics support.

And in between, lots of swell doodads like hard copy plotters and Videoprint from Image Resource, which produces a polaroid print of your screen image.

Siggraph '80 definitely has a lot to offer the "serious" attendee, although browsers were welcomed also. Siggraph (which is the Special Interest Group on Computer Graphics of the Association for Computing Machinery) knows what they're doing. Everything was coming off as planned and everything was planned well. Tutorials provided a get-involved experience for the computer literate with graphics tendencies or for those already involved in some unique aspect of the field. Since we spent only one afternoon at the show, it was difficult to gain an appreciation of the technical program. However, the session we attended (on Trends in High Performance Graphics Systems) presented top-notch people like Sproull and Fuchs, and was well paced and well attended. A fast but thorough glimpse of VLSI designs for graphics systems, it was aimed at the "machine people," but gave even the novice an idea of the research going on behind the fancy colors and rotating rockets.

Siggraph is an intelligent, special-interest show run by professionals, right down to the last pixel. Not an all-things-to-all-computer-freaks show, but one that delivered what it promised and should capture the attention of many personal computer end-users in the years to come. □

be seeing a feature-length film done entirely with computer animation in the near future. It's called "The Works", by the New York Institute of Technology, and the previews showed an epic space opera, including ships, aliens, battles, and robots. The graphics looked good and the story interesting; I can hardly wait to see it. New York Institute of Technology had a number of other entries; must be a fascinating place to work and play.

Another place we'll all be seeing some computer graphics is the new PBS series hosted by Carl Sagan, "Cosmos". There's about forty minutes of computer graphics in the series all together, such as showing how star birth and death keeps the spiral arms of the galaxies stable, and tracing evolution by transforming drawings from a one-celled organism to homo sapiens. The graphics were not the most advanced, but were useful in explaining difficult concepts; perhaps a good example of how computer image generation is moving out of the research labs and "gee whiz" commercials and into the daily life of the television producer and educator.

On the other hand, one entry tried to be entertaining and was billed as using computer graphics, but failed to deliver on both counts. Instead, it had various singers and disco dancers plus occasional special video effects which might have some computer involvement. These moviegoers were a rough audience; if they applauded loudly for arotating protein molecule,they hissed and booed when subjected to straight photography. I understand that this entry will be out on video disk, so be warned.

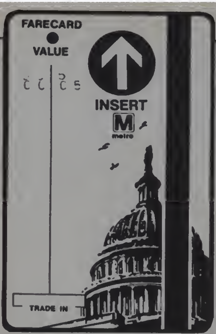
The name ACM always brings to my mind a group of computing machines, sort of like a labor union for robots.

Many of the entries were sampler reels from professional computer graphics houses, but nobody minded seeing variations of the same auto commercial ad or network logos, as long as the graphics were well done. An entry from the Soviet Union showed a simulation of a six-legged robot walking over rough terrain, apparently done by plotting every frame on paper. One of the organizers showed their "City and Flight" film, a humorous way of simulating travel by airplane and through an urban scene.

The film show was almost too successful; the hotel management threatened to shut it down because of overcrowding. The conference organizers managed to juggle 16 and 35mm projectors, plus a Light Valve video projector, with little delay between entries. One of the organizers, Rick Speer, and Bill and Ruth Kovacs, have compiled *An International Guide To Computer Animated Films*, containing descriptions and sources of over 200 films and tapes. It's available for $5.60 from Animation Research, PO Box 2651, Seattle, WA 98111. □

"It's nice, but I'd like to see it lay just one egg!"

©Creative Computing



DAWN OF THE DISC

David Lubar

First Encounter

The tunnel for the Washington D.C. Metro is a Disney-modern curving tube that seems calm enough until the wind comes, pushed by the red-line subway with an ear-popping rush. They've got a system here. Tickets. Magnetic tickets. You put cash into a vending machine, push some buttons, and take a card lined with a magnetic strip. The same card lets you out at the end of the trip. For me, the end was a highway-cum-moonscape in Arlington—Pentagon City—site of the conference on Interactive Videodiscs in Education and Training.

I met two persons from DEC at breakfast on Thursday. They'd been around and knew conferences the way a pool hustler knows nine ball. We killed an hour on trade talk—rumors of a new Apple graphics package, small talk about common acquaintances, and other trivia. Then, armed with a stack of blank tapes and other paraphernalia, I went in to catch the first talk.

The conference was a sedate affair, with a mostly-male crowd of 150 from a variety of fields. There was a group from ETS, men form MECC, front men from Sony, DiscoVision, and other manufacturers of video equipment, and members of the army. My last remnants of radical sixties anti-uniform paranoia were destroyed by the great work the Army is doing. More on that later.

Dr. Malcolm Davis, from the U.S. Dept. of Education gave the first address, "Impact of Videodisc Technology on Public Education." Among the points he made was the crucial idea that you can't put the same old material on video discs and think it's going to work. Revamped versions of old films just won't make it.

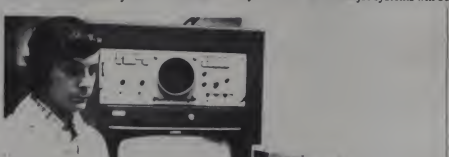
It soon became apparent that there is a good technology out there which just needs the right uses. But there are problems with this technology. Throughout the talks, the speakers honestly dealt with some of the problems and limitations. The discs, by themselves, could turn into nothing more than glorified movies. It was only when a computer got into the act that the magic happened. But some of the big miracles had nothing to do with discs. More on that too at a later time.

Among the issues covered during this contagious spate of honesty were problems dealing with a lack of standardization, with costs, and with technical difficulties. The industry is new, and the means of getting information onto a disc vary in large and small ways. There are capacitance methods, such as the one used by RCA. There are transmission techniques where the laser beam is read after passing through the disc, and reflection techniques using mirror-like discs.

Dr. Davis presented some hurdles facing videodiscs on their way into the classroom. School budgets were low, and teachers were reluctant to deal with anything that was spoken of in terms of replacement.

Throughout the day, the click of recorders running out popped through the room. What was needed, I realized, was a recorder that stayed off until something important was said, then recorded everything it missed.

The next speaker, Dr. Alfred Bork, added some more common sense to the air of the day. He believes the major systems will be



A phase in the mastering process.

Disc, continued...

of the stand-alone sort, and these systems will commonly appear in public places. He pointed out that you can develop an interactive video system in several ways; start with a program and add the video, start with video and add the program, or start from scratch. The third way seems best suited ot producing usable material. To tantalize us a bit, the Dr. also spoke of brain-wave input and the problems thereof.

John Volk slapped even more perspective on the issue by showing the capabilities and limitations of videodiscs. A disc that can store 54,000 frames, where each frame is equivalent to the image on a TV screen, holds a lot of data. But if that data is something that can be stored in a more compact form, the videodisc must cede its position to the hard disk. After all, you don't have to store an entire screen to store the lines of text that are on the screen. But hard disks drop back to second place when picture storage is needed.

Lunch was the next item, with a buffet available. I skipped it to investigate the pinball machines on the mezzanine. Sad to report, Arlington has followed the rest of the country in going to the outrageous quarter-a-game three-ball-per-game system. Is nothing sacred? After lunch I managed to speak with some of the others at the conference. Dr. Allen Glenn from the University of Minnesota, and from the Minnesota Educational Computer consortium filled me in on the work MECC is doing. They are presently working on integrating computers and videodiscs, producing educational programs in the field of economics. Dr. David Yens from the Mount Sinai School of Medicine also had some interesting comments. He explained that the interest in videodiscs was nothing new. The same thing had happened with educational radio (must have been before my time), educational television, and computers. People would predict major changes, others would worry about these changes, and nothing super-drastic would happen. Dr. Yens plays a major role in later less-academic portions of this story.

After lunch, Charles Frye took the stand. He is the creator of a CAI language known as Planit (see *Creative Computing*, vol. I, no. I). Dr. Frye believes that a common strategy for those doing videodisc programs will be to work the programs around existing pictures. Programming is called "authoring" those in the field. I will refrain from commenting on this coinage. Back to the issue, a series of lession where only thirty seconds was spent on each picture would mean a disc could hold 900 hours of visuals. There will be plenty of room on discs, and most will hold more material than any one instructor can use. That is good since the discs are rather expensive to create. At the moment, creation of a master costs around $10,000.

Testing the finished product.

Dr. Frye also spoke of the need to develop transportable software. Standardized techniques and languages are needed so any disc will run on any machine.

Mike Doyle from Thompson CFS came up next and told us all about his product. Thompson is working with Xerox on read-write discs and with 3M on mastering and replication techniques. Their disc uses the transmission method. The disc, made of polyvinyl chloride, is transparent. The laser zaps through it, reading either surface.

Dr. William Ford spoke about picture data-base systems, stressing the problem of assembling all the material and putting it on disc. He sees a need for a system that would tell how much room is left on a disc and where that room is. He envisions the creation of timesharing systems that will cost around 30K and allow instant access to over 100 discs.

Great Disc-coveries

So far, everything seemed interesting, but most of it also seemed to be in the planning stage. Bringing the audience back to the present, Dr. John C. Ittleson, from DesignWare, Inc., delivered a talk with the unpromising title, "Mentor Systems for Videodisc Authoring." The talk and demonstration turned out to be outstanding. The Mentor system uses videotape. While many systems put program information on the audio track, Mentor uses video frames. Seventeen seconds of tape can hold as much data as twenty Apple disks. This data is saved in a redundant manner to assure accurate storage and retrieval. The Mentor system uses videotape and computers to deliver lessons. But this is just the beginning. Or, more properly, the end. The lessons have to get on the tape. That could be one excruciating process. But the authoring system (there's that word again) makes it all a snap. The author can design a lesson with infinite (well, finite but great) branches allowing for questions and answers and all the other trappings of good CAI. Once the desired sequences are specified, the Mentor system creates the program and puts it on the tape along with the meat of the lesson. Dr. Ittleson mentioned that the system would also work with videodiscs, but they wanted to produce

SYBEX SPEAKS YOUR LANGUAGE . . .

THE PASCAL HANDBOOK by Jacques Tiberghien — A dictionary of every Pascal instruction, function, operator and reserved word covering virtually all versions of Pascal. 300 pp. 150 Ill. Ref P320, 7"x9", $14.95 ■

INTRODUCTION TO PASCAL (Including UCSD PASCAL) by Rodnay Zaks — A step-by-step introduction for anyone wanting to learn the language of PASCAL. Describes UCSD and Standard Pascals. 440 pp. 100 Ill. Ref. P310, 7"x9", $12.95 ■

INSIDE BASIC GAMES by Richard Mateosian — Uses a games format to teach program design in BASIC. Games run on TRS-80, APPLE II, PET/CBM and others. 300 pp. 100 Ill., Ref. B245, 7"x9", $13.95 ■

FIFTY BASIC EXERCISES by J.P. Lamoitier — Teaches BASIC by actual practice using graduated exercises drawn from everyday applications. All exercises written in Microsoft BASIC. 300 pp. 140 Ill., Ref. B250, 7"x9", $12.95 ■

THE CP/M HANDBOOK by Rodnay Zaks — Complete instructions and reference handbook for CP/M — the industry standard in microcomputer operating systems. 336 pp. 100 Ill. Ref. C300, 5½"x8½", $13.95 ■

PROGRAMMING THE Z80 by Rodnay Zaks — A complete course in programming the Z80 microprocessor and a thorough introduction to machine language. 620 pp. 200 Ill. Ref. C280, 5½"x8½", 2nd Ed. $14.95 ■

PROGRAMMING THE 6502 by Rodnay Zaks — Machine language programming of the 6502 from basic concepts to advanced data structures. 392 pp., 160 Ill., Ref. C202, 5½"x8½", 3rd Ed. $12.95 ■

6502 APPLICATIONS BOOK by Rodnay Zaks — Real life application techniques: the Input/Output book for the 6502. 288 pp. 207 Ill. Ref. D302, 5½"x8½", $12.95 ■

6502 GAMES by Rodnay Zaks — Third in the 6502 series. Teaches advanced programming techniques using games as a framework for learning. 304 pp., 140 Ill. Ref. G402, 5½"x8½", $12.95 ■

YOUR FIRST COMPUTER by Rodnay Zaks — The most popular introduction to small computers, what they do and how to buy one. 280 pp., 150 Ill. Ref. C200A, 5½"x8½", 2nd Ed., $7.95 ■

MICROPROCESSORS: FROM CHIPS TO SYSTEMS by Rodnay Zaks — Covers components, concepts and techniques from basic to advanced. 420 pp. 257 Ill. Ref. C201, 5½"x8½", 3rd Ed., $10.95 ■

MICROPROCESSOR INTERFACING TECHNIQUES by Austin Lesea, Rodnay Zaks — Hardware and software interconnect techniques including D to A conversion, peripherals standard buses and troubleshooting. 464 pp. 400 Ill. Ref C207, 5½"x8½", 3rd Ed., $15.95 ■

PROGRAMMING THE Z8000 by Richard Mateosian — Architecture and function of the Z8000 and its family of support chips. Includes programming in Z8000 machine language. 312 pp. 124 Ill., Ref. C281, 5½"x8½", $15.95 ■

CIRCLE 199 ON READER SERVICE CARD

MAIL TO: SYBEX
DEPT. CC1
2344 SIXTH STREET
BERKELEY, CA 94710
Phone Orders: 415/848-8233

SYBEX

NAME ____________ ☐ SEND ME YOUR FREE CATALOG
ADDRESS ____________
CITY ____________ STATE ____________ ZIP ____________
ADD ☐ $1.50/book UPS or ☐ 75¢/book 4th class mail (CA add tax) Total Amt. Enclosed ____________
OR CHARGE MY ☐ VISA ☐ MC ☐ AM EX. CARD NO. ____________
SIGNATURE ____________ EXP. DATE ____________

Disc, continued...

a system usable with available technology. Mentor was very impressive, and I expect to see a lot more in the way of innovation from Dr. Ittleson and DesignWare.

The day was capped off with another winner. Dr. Steve Levin demonstrated videodisc-based surrogate travel (you can get there from here). It was great. What they do is go somewhere and film everything. Then it's all put on a disc. Next, someone sits in front of two CRT's and uses a joystick to move around on a map of the place. Once an area has been selected, he switches to the other screen and takes a walk through the scene. The joystick controls speed and direction of travel. The angle and height of viewpoint can be changed. Great. Imagine the adventure games you could have with this system. (Are you listening, Scott Adams?) And, wonder of wonders, there are even practical uses. Dr. Levin showed an example of a disc made of San Francisco Bay. The pictures were taken from the same height as the vantage point of a ship's pilot. The disc could be used for training. A budding young pilot could get to know the waters with no risk of sending himself and a million dollars worth of ship to the briny deep.

Very impressive. And the best part is the time factor. The shooting of the harbor only took seven hours. They also did a nuclear reactor in three hours. Obviously, they've got the system down cold. This is a great use of videodiscs, making full use of the abilities of the medium instead of merely transporting old ideas to a new form. The system is presently running on a Z-80, but they are reconfiguring it to run on an Apple.

That wrapped up the business end of the day. Everyone moved to another room for a cash-bar, pretzels-and-chips reception. I mingled for a while, spending most of my time answering questions about Sesame Place, which many people at the conference wanted to know about.

There was little talk concerning videodics. Everyone probably needed a break from the subject. No hot scoops, no industrial secrets. Oh well.

Food and Peril

After the reception, Dr. Yens and I decided to go in search of nourishment. We had an interesting ride in search of a Vietnamese restaurant, getting lost on the way and ending up at the Pentagon. (It really does have five sides.) The restaurant was closed. On the way to a second restaurant (of the open variety), Dr. Yens told me some of the work he had been doing with computers, and some of the ideas he had for future projects. While this was all interesting, I missed a lot of what was happening while alternating between spates of fear and admiration for the Doctor's driving. He handled his Plymouth like he'd been born with the wheel in his hands. Sort of the same beauty that can be found in a good horror movie.

At the restaraunt, Dr. Yens went into more detail about the similarity between educational videodiscs and educational radio. He seemed to represent the popular attitude at the show: these things are interesting and powerful, but what can they do and what difference will they make in education? Like most of the attendees, however, he seemed to be impressed by some of the applications. After dinner, we went to the piano lounge at the hotel, where I listened to an excellent singer and drank some decent concoctions from the depths of a blender while the good Doctor wore out several partners in a frenzy of disco fever.

Back to the room, I began drafting some ideas for this article. As should be obvious, the two things that impressed me the most were the Mentor System and the surrogate travel. One ran on tape but could be adapted to disk; the other required a disc. Videodiscs weren't going to shake the earth or change the face of education, but they would have their uses, they would allow some new approaches to education. Like the little magnetic strip on the subway card, like the cassettes in the tape recorders, videodiscs were just a way of storing and retrieving information. A way that served well in certain situations, but should not be used in others.

This is a great use of videodiscs, making full use of the abilities of the medium instead of merely transporting old ideas to a new form.

Flip Side

This is getting a bit long, so I'll just go into the highlights of the second day. Mark Heyer from Sony demonstrated the disc his company produces; it is stamped, one sided, and compatible with the DVA format. The player employs a moving lens and steady spindle instead of the usual moving spindle approach. The player can store 500 commands.

Dr. James Baker from the Army (ours), took the podium and explained how the military was drowning under a paper deluge. There were tons of field manuals and other books being used, and the situation was reaching a critical point. They see videodiscs as the anser. In cooperation with Charles Frye, they are developing a version of Planit that will run on small computers. The plan is for this language to be transportable. They want it to run on any small computer. A neat trick, if is possible. The goal is to develop a sturdy field unit consisting of computer, videodisc player, and screen; perhaps something the size of a portable stereo. This could be carried on maneuvers. The discs would contain a variety of information; pictures, programs, practice games, and a dual audio channel. Perhaps the Army and the paper flood can do for the videodisc what NASA and the space race did for the computer.

It was getting late and my train would be leaving soon, but the Army had me hooked, and I stayed around to hear Colonel John Goetz. He explained that the videodisc had to be more than just an electronic page-turner. One possible use was in electronic training. With the aid of a light pen, trainees could attempt to patch circuits that were displayed on the screen. The disc would display different parts of the circuit, as well as meters and whatever. The Army, by the way, is planning to use Apples for some of these projects.

It was time for me to head back through the labyrinth, armed with another magnetic fare card and a new respect for the potential of the videodisc—a respect tempered with the realization that, amidst the few innovative and appropriate uses, the discs will probably be used also in all the wrong ways for all the wrong applications. But they should survive this treatment.

Yeah, they use magnetic strips on little cards beneath the streets of Washington. And the card also makes a great book mark. But I wouldn't try to store a program on that little strip; it just isn't designed for that. □

For further information about Videodiscs and computers, see *Creative Computing*, vol. 2, no. 2. An article by Alfred Bork discusses the role of an "intelligent" stand-alone videodisc system in education while a short piece by Arthur Luehrmann talks about a $2.98 computer library. After all, one videodisc can store 10^{10} bits of information. Other pieces discuss the technology of videodiscs, future technology and artificial intelligence. The issue is available for $2.00 postpaid.

Scientific American, Vol 243, Number 2, pp. 138-148, contains a good article on disk technology, covering both magnetic and optical techniques.

The proceedings from the conference on Interactive Videodiscs in Education and Training will be available in the near future. For information, contact the society for Applied Learning Technology, 50 Culpeper St., Warrenton, VA 22186.

Video Product News, P.O. Box 1069, Los Angeles CA, 90028, is a new bimonthly magazine which reviews new products in the home video field. A one-year subscription costs $24.00

A Visit to SESAME PLACE

Betsy Staples

An array of cargo nets, strung four-stories above the ground, is just one of several ways of circumnavigating Sesame Place. Slat bridges, ramps and the ground are among the other routes in Langhorne, Pa. throughout the park

Sesame Place, billed as "an innovative play park for children three to thirteen offering a free-flowing combination of outdoor physical activities and challenging educational games," opened to the public on July 30, 1980, only three months after its scheduled May 1 opening.

On July 29, we gathered up Kristin, a nine-year old niece, and Gordy, a three-year old nephew, and set off for the special preview opening of Sesame Place. It was pouring rain when we left, but we decided to take our chances.

By the time we arrived in Lower Bucks County, Pennsylvania an hour later, the sky had cleared and the day was hot and steamy. We were met at the gate by young staff members uniformed in the yellow, red and green theme colors of the park. Each of us received a commemorative packet containing a large button, a patch and a sheet of stamps all sporting the park's Big Bird logo.

And not far from the entrance was the Bird himself singing and cavorting above the crows on (could it really have been?) a *cat* walk. The children found him amusing for a while, but soon wanted to try some of the activities.

Kristin's first choice was a climb through a web of cargo nets strung four stories above the park. Gordy was game, but his little feet slipped too easily through the mesh, so he settled for a few trips down a giant orange and purple plastic slide.

In the Sesame Street Construction Company area, Gordy required a bit of help from another little boy to lift the lightweight, interlocking polyurethane blocks into position to create the foundation of what he claimed was to become a garage. Both children found the Monster Maze, a forest of six-foot high punching bags, a total bore. Apparently others did, too, since it was the only functioning attraction in the park completely devoid of activity.

Emphasis throughout the park is on participation, and participation is made easier for younger children by helpful high school and college age staff members — both male and female — who are alert to trepidation and happy to provide support when needed.

Unfortunately, many of the 40 "outdoor play elements" were out of commission, awaiting, we supposed, final adjustments to make them playable.

The children wanted to try the Count's Ballroom, a pool of 80,000 green plastic balls in which to dive, swim or just disappear. However, the weather had taken its toll, and the 80,000 green plastic balls had been joined by many gallons of rainwater. We watched as staff members tried a pump (water not deep enough) and finally a vacuum cleaner to remove it. Seeing the antics of the young vacuumers whetted everyone's appetite for a romp with the Count.

Gordy was in the first group invited in. (Children are separated by size into two groups which alternate periods of play.) He slid into the pool and stood stock still as the sea of green balls closed in over his head. An alert helper rescued him, but he was not eager to try again. Kristin was apparently just the right size, because she had a ball.

After trying all the outdoor attractions, we went in to the Computer Gallery. As regular readers of *Creative Computing* know, we were involved in the early stages of the development of this area, but we

Gordy's sneakers bring his trip down a giant purple and orange slide to a temporary halt.

Sesame, continued...

were surprised to see the transformation that a room full of Apples had undergone.

Each machine consists of a touch-sensitive keyboard with large "keys" arranged in alphabetical order. There are also arrows for moving the cursor, and additional keys as required for special games. Conspicuously absent is the "reset" key.

The color monitor is housed in a large, brightly colored metal box which also accommodates the "coin-op" mechanism. Tokens are required to change the pro-

Throngs of adults mob the computer center at Sesame Place to escape an afternoon thunderstorm. tokens were free at the press preview but normally are three for $1. Each token provides 4 minutes of play.

Big Bird serenades his fans.

gram from attract to play mode. They are on sale at three for a dollar.

Players sit on movable, carpet-covered boxes, and the designers seem to have done a good job of child-proofing the machines and their surroundings.

All the Apples are controlled by a Nestar system temporarily (we hope) housed in the staff men's room.

Less than half of the advertised "70 specially created electronic games" were available in the Gallery when we were there. Some of the more clever ones included Reflect, in which the player tries to bounce a beam of light off a mirror at exactly the right angle to illuminate an object, and a non-violent version of Hangman with amusing graphics. In Mup-O-Matic a picture is created on the screen as random pixels are filled in. The child presses a button to stop the picture and type in his or her guess as to its identity. This was fun for a few times, but grew tiresome in its myriad variations — Muppets, fruit, animals, sports equipment, etc.

Kristin was able to enjoy most of the games, but since almost all of them require that the child be able to read, Gordy was not impressed.

At the Sesame Food Factory, we were treated to a cheese and sprout sandwich on a large soft pretzel, raw vegetables, a plum and a peanutbutter brownie in a box lunch followed by ice cream cones. The menu includes some unusual and tempting items, all of which promise "a minimum of preservatives and additives and appreciably lower amounts of sugar and salt."

What was our overall impression of Sesame Place? It is certainly a colorful, comfortable, interesting place to visit — particularly for children. There is much to be learned and the children we took had a grand time, even though Gordy, at the bottom of the suggested age range, was

A giant screen repeats the Mup-O-Matic picture on the monitor. "Please type in your guess now."

The brains of the outfit: this Nestar system controls all the Apples in the Computer Gallery.

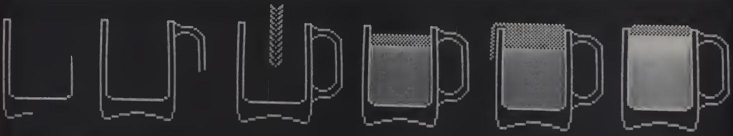
barely old enough to enjoy some of the activities and just plain too little to use others.

Our main disappointment was in the fact that there is virtually nothing for adults (anyone over 5'2" or thirteen years of age) to do. Of course, it's fun to watch the children having a good time, and a few of the computer games are suitable for grown-ups, but other than that parents, teachers, older siblings, aunts and uncles will find that their most important role at Sesame Place is keeping track of socks and sneakers while the little ones bounce, build, crawl, climb and swing their way through the park.

No, those are not Lilliputians wallowing in fresh peas; that's Kristin in the foreground enjoying a romp in The Count's Ballroom.

Players sit on carpeted cubes and type on specially designed keyboards to try the games in the Computer Gallery.

Sesame Place is intended to appeal to residents who live in the surrounding geographical area rather than to tourists, and is said to be designed for many visits of two to three hours each. This is a nice concept, but we doubt that many parents will be willing to pay the $3.95 admission fee for themselves on a regular basis. This park seems to be the perfect example of a place where an adult accompanied by a child should be entitled to a substantially reduced rate. □



Creative Animation

THE CROWD STOPPER

David L. Ross

What do a popcorn popper, diesel engine, birthday cake, beer mug, and a corporate logo have in common? Not much, you say? No so. These items, and countless others, can be computer-animated for display on a color TV screen or monitor in a visually fascinating fashion.

But why would anyone want to animate a popper or beer mug, I hear you cry. It's an area of advertising for which the waters are virtually uncharted. Because people enjoy watching the creative process of image development on the screen, computer animation is an effective promotional tool. Whether the display is placed in a trade show exhibit, store window, building lobby, or permanent product display, the results are impressive. The creative motion on the screen attracts a crowd in a way that a static display or sign seldom does. If cleverly done, animation communicates information to viewers in an entertaining, colorful way. In effect, it's a localized "TV spot" that runs continuously and grabs the attention of passers-by—at a fraction of the cost of regular network or videotape alternatives.

We recognized the universal appeal of the TV screen in early 1979 and began to investigate the possibilities inherent in the use of computer-controlled message displays. At that time, the closest application of this nature was the continuous scrolling of text commonly seen in hotel lobbies and other public areas, announcing meeting rooms, schedules, etc. Such displays are primitive—they're visually boring and can easily be ignored by viewers. Recognizing the potential of and need for more visual impact, we began developing graphic displays to communicate key points, using text only where necessary. Today, our presentations are 70-80% customized color graphics.

We're building a library of animation sub-sequences that can be "dropped into" larger productions—speedboats, champagne glasses, iced cakes with burning candles, flags. We're also developing software vehicles that make program customization, such as the inclusion of corporate logos, a simpler process. A further extension of this concept is the marriage of the video game and advertising presentations, in which the viewer actually participates in the exhibit. This is becoming a powerful "draw," particularly in trade shows. Promotional messages related to the game in play and/or the exhibitor's products and services can be entered from the keyboard, allowing "instant customization," or embedded directly in the program. The originality of this advertising concept, as well as the portability, reliability, and cost of small computers makes the Crowd Stopper an attractive alternative to videotape players or other traditional display devices.

The Popcorn Pumper—Custom Animations

Our popcorn popper animation is summarized in eight frames taken from an animated presentation developed for Wear-Ever Aluminum, Inc., a subsidiary of Alcoa, for trade show display use. Like most animations, it relies heavily on the impact of imaginative development on the screen; it has to be seen in action to fully appreciate the effects. The Popcorn Pumper is drawn on the screen in two colors—yellow and white (just like the real product)—on a black background. The base and chamber outline are built at a variable rate, with

The importance of image development is illustrated in the popcorn popper animation sequence.

David L. Ross, President, Micro Video, P.O. Box 7357, 204E. Washington St. Ann Arbor, MI 48107.

accompanying musical tones. Then we add the frenzied motion of popcorn actually "popping" in the chamber, complete with sound effects, and spill it out into the waiting bowl. Viewers invariably chuckle over this part of the presentation, and stay to watch the image development a second, third, ... time.

The popcorn popper was one of four products animated in a single program for use in Wear-Ever's Housewares Show exhibit. The presentation achieved its goal, stopping traffic in the aisle and creating interest in the Popcorn Pumper and other products Wear-ever manufactures for the home. It was so successful that the company has reused it numerous time in other trade shows.

Animations—Getting Started

To produce an effective animated display, we follow the procedures generally used in preparing speeches and written materials, with minor adaptations:

- Understand your audience. Who will primarily be viewing the presentation? Are you trying to attract the attention of adults or children? Men, women or both?
- Define your objectives. Do you want the animated presentation to sell the product or service and provide technical information as well? Or do you wish to simply stop and entertain the passing crowd? If so, for how long? What interaction with viewers do you want to effect? Will an "animated billboard" meet your goals? Or, do you want to use interactive advertising—customized video games with interweaved promotional messages—to invite participation in the exhibit and hold the viewers' interest long enough to allow company representatives to make personal contact?
- "Storyboard" the entire presentation. Try to achieve balance in the visual material, keeping it directed toward the anticipated audience and defined objectives.
- Develop the graphic images, transitions between images, and other highlighting effects, varying the animation techniques for high visual impact.
- Review the production with the client, and expect to make modifications and improvements. Seldom, if ever, is the first production the final version. Improvements can always be made.
- When approval of the presentation is final, make arrangements to watch the presentation in actual use. Is it effective? How well does it compete for people's attention in the environment in which it's being used? What portions are visually dull and need to be improved? Is the presentation meeting your defined objectives? Make note of needed modifications for future use.

An interesting phenomenon in computer-animated displays is that people's attention is generally held over multiple viewings. If you watch people in the vicinity of a presentation, you'll see their eyes continually drawn back to the screen—no matter how many times they've seen it before. You can use this as a barometer of the presentation's effectiveness.

Rules of Thumb for Successful Animation

We've evolved a set of guidelines that we believe differentiate good animated presentations from poor ones. While there are exceptions, these "rules of thumb" generally yield effective displays:

1) *Never scroll text vertically on the screen.* The human eye does not easily read material presented in this manner. More effective methods include partial screen wipes, erasing material by overwriting it in the background color, etc. The speed at which text is output to the screen and length of time it remains there are also important factors.

2) *Keep in mind that the method used to put the image on the screen is more important than the end result.* If the reverse were true, then a photograph or videotape of the actual product would be sufficient, as well as more technically accurate. Animation has the appeal of the quick sketch cartoonist at an amusement area. He holds his viewing crowds while he's drawing the picture, but tends to lose a large portion of this crowd and potential customers upon completion of the sketch.

3) *Always keep motion on the screen.* Avoid totally static screens by including at least slight movement with color changes, flashing, moving stick figures, etc. Motion ensures that viewers' eyes will stay glued to the screen—they want to see what will happen next.

4) *Avoid painfully slow image development, as it strains the viewer's interest in the display.* Faster graphics can be achieved by using broader lines or simplifying internal calculations that produce the image.

5) *Use unusual sequences to draw images.* This provides an element of surprise in the presentation and creates suspense. For example, if you need to draw a rectangular box, rather than using a single continuous line, consider drawing pairs of parallel lines going in different directions simultaneously. Or, if you want to include the American flag in your animation, don't draw it one stripe at a time—that's far too obvious. One approach you might use is to draw all red stripes simultaneously in one direction, followed by all the white stripes in the opposite direction, then add the blue field and output stars in a seemingly random fashion. The idea is to create suspense and pique the viewers' interest in whatever image you're producing.

6) *Vary the speed of the presentation.* Depending on the image and effects desired,

Crowd Stopper, continued...

vary the speed accordingly. Don't draw everything as fast as possible—but allow the viewer to savor the image development. On the other hand, don't let a portion of the presentation drag enough to evoke a visual yawn.

7) *Use color for emphasis*. Color-code key concepts to improve viewer comprehension of the material. But don't carry this too far and use too many colors on a single screen simply because the computer has the capability of doing so. "Color overload" is as poor as a dull black and white presentation.

8) *Choose color combinations carefully*. Use colors that match the objects of animation if possible. Vary the color combinations throughout the presentation, but make sure all combinations are aesthetically pleasing, taking into consideration the audience, locations, and objectives.

9) *Use sound effects, if available on the computer, to highlight the animation*. In our beer mug animation, for example, sounds accompanying the graphics simulate the sounds of beer flowing into a mug, varying with the rate of flow and the fullness of the mug. Try to imitate sounds as appropriate to the image, but don't use a lot of non-related sounds, because in time they become annoying rather than entertaining. Also, don't continuously use sound for sound's sake in the presentation. Sounds add another dimension to the presentation, but too much sound will actually reduce, rather than augment, the impact.

10) *Use variety in message display*. experiment with three dimensional lettering, oversized letters, word swimming, color rolls, fades, and wipes. Words can be "shot out of cannons," color-highlighted, or "spoken" by animated characters or stick figures. Be sure that the text is readable by viewers, considering their distance from the screen, the screen size, lettering size, and duration held on the screen. You might also program the capability, as we do, of message input from the keyboard to allow last minute changes in presentation messages. This adds futher flexibility.

11) *Add humor to animations*. If your audience laughs, they'll watch longer. People love to be entertained.

12) *Pay special attention to transitions between graphic images*.Don't always clear the screen before producing another image. Allow one to evolve into another some of the time.

13) *Don't rush*. Good animations take time. Experiment with different ways of producing a single image and determine which is most visually effective, as this invariably leads to a better final result.

14) *Don't prejudge or limit your animation possibilities*. Life can be imposed even on lackluster objects such as a frying pan or toaster by the way they are drawn and the addition of a flickering flame beneath the pan or toasted bread "popping" out of the toaster.

A cardinal rule of animations is that they improve with experimentation and experience, provided enough time is allowed to do the work. Did I say "Work"? Yes, there are hours of work in every animation, but it's one of the most imaginative applications for home computers, and can give more sheer pleasure and satisfaction than other types of programming. It's enormously gratifying to watch people become enthralled with and chortle with delight over a particularly clever presentation.

So, consider the possibilities! Almost anything can be animated. We use the Interact computer, Microsoft Basic, and machine language to produce most of our animations. However, it's not so much the computer or language that you use, but the imagination that you put into programming the display that makes the difference. Give animation a try on your own computer. We think you'll find it will open up a whole new world of programming enjoyment and creativity for you. □

Artist/Computer Dialogue

Bernard Demio

Reprinted from PAGE, the publication of the Computer Arts Society. For membership information, write Kurt Laukner, Math Dept., Eastern Michigan Univ., Ypsilanti, MI 48197.

In his creative approach the artist uses a mini-computer in the following manner: (see diagram).

First Stage:

Starting from a basic idea, the artist does some preparatory conceptual work in the form of drawings, rough sketches, and various other attempts. He defines the constraints of his creative work and that which he wants to express or omit. (For example the definition of the initial generating forms).

Second Stage:

Setting up the formal computer language for these forms, the colors and the controls for the envisaged composition. Eventually returning to certain elements defined during the first stage.

Third Stage:

Artist/mini-computer dialogue. Given a work program adapted or chosen by the artist (program of forms or colors, in low or high resolution), the computer will establish the relation between the data of the composition defined during the first stage, and this work program.

The program will process the data and suggest combinations to the artist.

As a function of his investigations, the artist will be able to explore one path of testing rather than another.

In exploring a path, the artist will have new ideas which he may or may not reintroduce as data. He will think of new composition controls which will enable him to quickly put into effect his ideas. He will then be able to decide whether he will keep the results of his testing path.

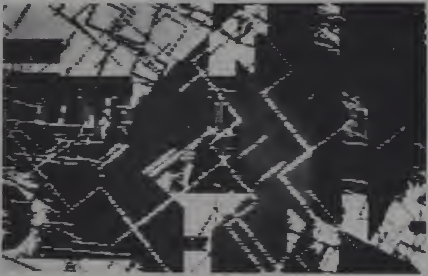
If he is not satisfied, he will pass on to the exploration of another testing path.

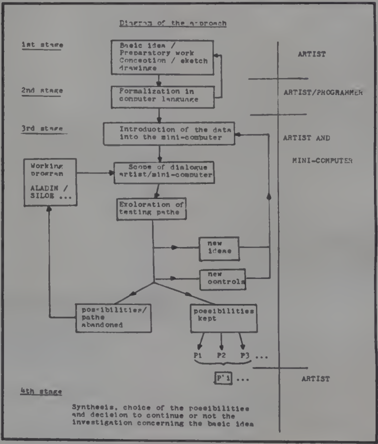
At the end of the processing/dialogue, he will thus have for his investigations one or two possibilities corresponding to his creative controls defined in the first stage: or modified along the way.

Fourth Stage:

A synthesis by the artist of the different possibilities issuing from the 3rd stage, and the choice of one possibility or the continuation of the investigation. □

Bernard Demio, 12 Rue Rambuteau, 75003 Paris.





Build Your Library!

Our warehouse is bursting at the seams. We just don't have enough room for everything we publish. So we've put together some cartons of every single thing we print plus some other books and Cut the price nearly in half. The books are brand new—not damaged or rejects. The magazines range from July/August 1977 to July 1980. We don't have every issue, of course, but there's over 3600 pages of good reading in the 25 issues of Creative Computing and another 800 pages in the 8 issues of ROM.

Huge 45% Discount!

Here's the ideal way to start or build a basic computing library. There are 16 books on computer literacy, computer games, educational applications, the impact of the computer on society, sets of problems and much more. The package has 33 timeless magazines. You'll get a nifty board game, an LP record, nine posters, and a set of binary dice. All together 63 separate items for only $120.00 postpaid!

If you just want the 33 magazines alone, they're available for $41.00 postpaid, still a whopping 42% discount.

Orders must be prepaid by check or money order. (Sorry we can't afford WATS charges or bank card charges on a deal this good.)
Send order and payment to:

Creative Computing
Sensational September Sale
P.O. Box 789-M
Morristown, NJ 07960

Limited Time Offer

Expires Feb. 28, 1981

Here's what you get—

16 Books	List
The Best of Creative Computing, Volume 1	$8.95
The Best of Creative Computing, Volume 2	8.95
The Best of Byte, Volume 1	11.95
Basic Computer Games	7.50
More Basic Computer Games	7.95
The Colossal Computer Cartoon Book	4.95
Games With the Pocket Calculator	3.95
Computer Coin Games	3.95
Be A Computer Literate	3.95
The Home Computer Revolution	2.00
A Guided Tour of Computer Programming in Basic	6.15
Computers in Mathematics: A Sourcebook of Ideas	15.95
The Impact of Computers on Society and Ethics: A Bibliography	17.95
Katie and the Computer	6.95
Problems for Computer Solution-Student Edition	4.95
Problems for Computer Solution-Teacher Edition	9.95

Save!

Miscellaneous Items	
Computer Rage Game	$8.95
Binary Dice (Set of three)	1.25
First Philadelphia Computer Music Festival 12" LP Record	6.00
Computer Myth Posters (Set of eight)	3.00
Mr. Spock Poster	1.00
Reprint: Sorting, Shuffling and File Structures	.50
Reprint: Word Processing	.50

33 Magazines	
Creative Computing (25 issues from Vol. 3, No. 4 to Vol. 6, No. 7)	$50.00
ROM (8 issues, numbers 1 to 8)	16.00

SubTotal	$213.20
Less 45% discount	95.94
SubTotal	117.26
Postage and handling*	2.74
Total	$120.00

*Foreign postage $15.74 —Total $133.00

Permutating Lines

Torsten Ridell

The images reproduced here are some examples based on my idea of "permutating lines".

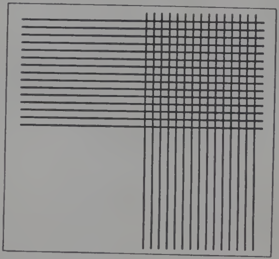
On a given surface, there are both horizontal and vertical positions for lines. These lines are distributed either systematically or aleatorically, resulting in a series of two-dimensional images.

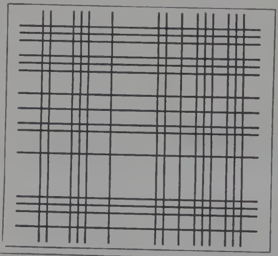
I next tried to combine some of these series to create a three dimensional image.

Each of the serial progressions of lines was used as a section through a cube where the distance between each line is equal to the distance between each section. By rotating each cube on its axes, I obtained a new set of drawings whose serial progressions are in three dimensions. (opposite).

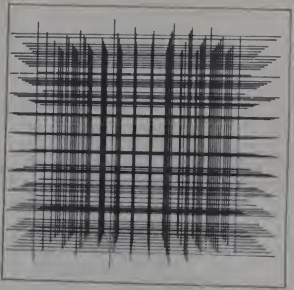
For the earlier series of drawings (two of which are shown here) I used the machine for its capacity to make repetitive movements, in order to achieve the desired precision. For the later series, of which three are shown, the computer itself played a determining role: it allowed me to "see" my ideas as concrete visual forms, giving rise to some highly complex images which I would have found difficult to realize by traditional methods, or of which I might never ever have conceived.

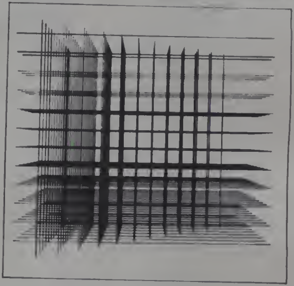
It is for this reason that I regard the computer not simply as a useful tool, but as being complementary to my artistic creativity. □

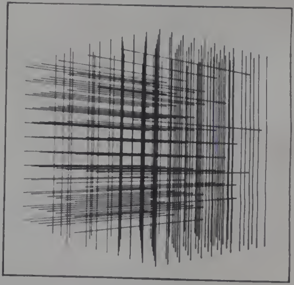




Torsten Ridell, 754 Rue Charlot, 75003 Paris.



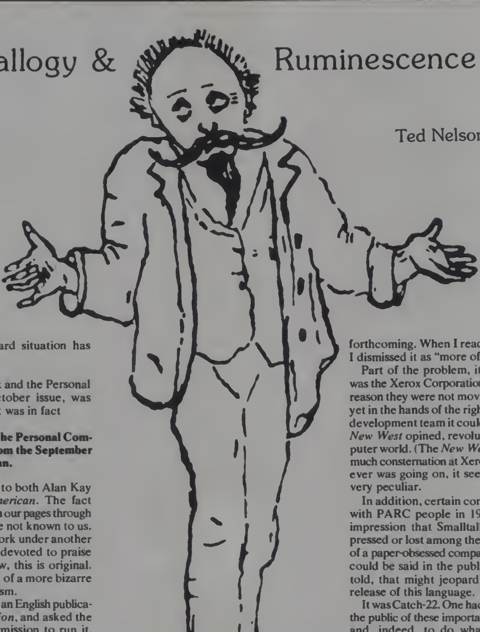




Reprinted from PAGE, the publication of the Computer Arts Society. For membership information, write Kurt Laukner, Math Dept., Eastern Michigan Univ., Ypsilanti, MI 48197.

Appallogy & Ruminescence

Ted Nelson



An extremely awkward situation has arisen.

The article "Smalltalk and the Personal Computer," in our October issue, was incorrectly identified. It was in fact

"Microelectronics and the Personal Computer," by Alan Kay, from the September 1977 Scientific American.

We hereby apologize to both Alan Kay and The *Scientific American*. The fact that this piece appeared in our pages through plagiarism was of course not known to us. But to reprint a man's work under another name in a issue largely devoted to praise of the real author—now, this is original. We have not ever heard of a more bizarre and astounding plagiarism.

We found the article in an English publication, *Computer Education*, and asked the putative author for permission to run it, which he gave.

It turns out, though, that for no known reason he had retyped the original and submitted it to the English publication. What possessed him to do it, let alone give us permission, when he might have gotten away with it, no one knows. (I called the culprit and asked him, and he didn't know.) He may lose his job for it.

We need not dwell on this unfortunate individual and his curious acts. What remains to be explained is how *we* failed to recognize the piece.

I read it, of course, when it originally come out; and like some other sophisticated readers I have talked to, I found it woolly and unhelpful. What was the *structure* of Smalltalk, this marvelous language, I wanted to know; but the examples in the article were unfathomable, and all the talk about children was off the point.

Many people had by 1977 seen the Smalltalk language and its marvelous capabilities, either as guests at Xerox Palo Alto Research Center or in the movies their researchers showed at conferences. Everyone was awed; many were frustrated by the lack of information on the language itself, or why it was so special.

My frustration ran deeper than most people's. I had been a guest of Xerox PARC as early as 1972, and had considered Kay's answers to my questions evasive and not forthcoming. When I read the 1977 piece, I dismissed it as "more of the same."

Part of the problem, it seemed to me, was the Xerox Corporation itself. For some reason they were not moving on this thing; yet in the hands of the right marketing and development team it could, as an article in *New West* opined, revolutionize the computer world. (The *New West* article caused much consternation at Xerox PARC.) Whatever was going on, it seemed to me, was very peculiar.

In addition, certain conversations I had with PARC people in 1978 gave me the impression that Smalltalk might be suppressed or lost among the corporate gears of a paper-obsessed company. Thus nothing could be said in the public presses, I was told, that might jeopardize the eventual release of this language.

It was Catch-22. One had a duty to inform the public of these important developments, and, indeed, to do whatever might be possible to help bring them forth; yet all this should be done on tiptoe, lest the slumbering paper giant find out the paper was threatened, and bury this wonderful thing in an unmarked grave.

Besides these problems, I had a great personal desire to understand it all, and then explain to those readers who benefit from my explanations what this was all about.

A number of breaks occurred in 1979. One was the appearance of a technical paper on the internals of Smalltalk—a sign, I had been told, that Smalltalk might eventually be released. And two fellows from Texas published a paper on their own version for the Z80. Moreover, I became editor of this magazine, giving me time and telephones to probe into these matters.

So a special Smalltalk issue was projected,

first for October, then for August, then October again.

The original idea was to do a special issue on Smalltalk itself. But it had to be done without the cooperation of Xerox Public Relations; dealing with them before had been a waste of time, and they, too, appeared to be covering something up.

The objective of the special issue was to get at the parts of the Smalltalk idea that the Xerox people were not publishing—especially the syntax (or lack of it), and the ways that Smalltalk functions are defined and redefined to give you the custom language you actually want, even as your idea of what you want is changing.

However, I did mention these plans to the PARC people, and got the old story: "Don't jeopardize what we're doing, we're trying to get the language out." So the piece was broadened to "actor" languages in general, and I omitted the most important statement that should have been made, although the astute reader may have seen it twinkling at him between the lines:

Smalltalk is the most important computer language anywhere, and will probably revolutionize the computer world.

As it happened, our October issue come out just after the meeting of SIGSMALL/SIGPC in Palo Alto, where Adele Goldberg of Xerox PARC announced that the Smalltalk language will in fact be put in the public domain by Xerox with the publication of a many-authored book on it, sometime in mid-1981.

And so, in the ringing phrase of the 1940s, Now It Can Be Told.

Smalltalk is the most important computer language anywhere, and will probably revolutionize the computer world, and the rest of the world as well.

In any case, when Dave Ahl showed me the Alan Kay piece, as reprinted in *Computer Education* over a different author's name, I still didn't like it. It seemed to me woolly and vague. But I tossed it on my heap of technical papers on Smalltalk and actor languages.

The October issue gradually took shape around my piece and others, explaining both the history and inner structure of Smalltalk and the other actor languages. I had learned a good deal, especially from MLT people and papers. And I picked up *Computer Education* again a couple of times, and began to see something in the piece; especially since I now knew what actor languages were about and how Smalltalk functions are defined and redefined. Browsing, I thought, Hey, this is pretty good.

So we asked for permission.

And as I did the preliminary layout on the article I read it again and thought, "Gee, this is *really* good, why haven't I heard of this guy before?"

Any reader knows that a thing changes between readings. If you read a book or an article one year, and then again five years later, it may say something entirely different to you; your mind is differently prepared, meanings and qualities and points stand out that hadn't before, and perhaps the earlier overall impression has changed because of what you have seen in the meantime.

So it was that I failed to recognize Kay's article. It took several readings, *and* a general understanding of the language's unique inside-out, upside-down structure, before the depth and wit of his words got through to me. Proving once again that Alan Kay is far ahead of us all.

So after much fuss and bother I finished with the two-part actors piece and took a trip to the Far East. And when I returned I found out we had botched it, and offended Dr. Kay, for we had indeed run his very own original article.

The editor's nightmare had come true. We had published a piece submitted fraudulently by someone other than its author. (The fact that it was not submitted to us, but that we found it in the plagiarized version and failed to recognize it, is just more complication.)

Well, it all got sorted out, and I had some very agreeable chats with Alan Kay and Adele Goldberg and the publisher of the *Scientific American*, and they were all very nice, and the matter is done with. So all's well that ends well. (Except for the culprit, who pretty much asked for what he got.)

Both Dr. Kay and the publishers of the *Scientific American* have been very understanding of our position in this affair. Kay's reputation certainly has not suffered, nor has that of the *Scientific American*. For us it's a different story.

In that glorious new volume, *The Next Whole Earth Catalog*, which came out about the same time as our October issue, it says that *Creative Computing* sometimes doesn't check its sources and publishes plagiarized material.

How could they have known, given that they went to press before we did?

And, realistically, how in the world can a publication protect itself against plagiarized submissions—especially if the editors *haven't* seen the piece to begin with?

It's just a risk we take if there are to be magazines.

* * *

"Those Xerox PARC people—they're so stuck up," someone said to me at a party recently.

That's not it, exactly.

They're on the other side of the looking glass, and there's too much to explain, so they speak in playful terms and Sufi parables.

I hope that the October and November issues, in their small way, may have reduced the explanation gap, so that others can better understand the Delphic sayings of the PARC people.

There is no question that one of the two or three most important places in the computer world is Xerox PARC. We will continue to follow their doings and influence in these pages, I hope now with a little more understanding on both sides.

There is much I disagree with in their approach and views of what people need. I think they were wrong, for instance, in waiting so long, and perfecting and perfecting, before putting Smalltalk out to the public. While the team's glacially-slow perfectionisn has produced a magnificent product, I would argue that the world would be a much better place now if we could have had their discards five years ago. (An Apple with the Rosetta language—representing approaches PARC discarded—would more than meet my own day-to-day programming needs, and do so far better than Basic.)

It comes down to what you think of the future. If we had decades to fool around, well and good—but with both thermonuclear war and global famine on the horizon, time lost is lost indeed. Time is short; we are a lot closer to Armageddon; and the better future through personal computing is still almost as far away as it was when the Altair came out. (The hardware turkeys are still gobbling up circuit specs, and people are still using idiotic words like "microcomputer". But there is at least one reliable keyboard machine with disk, and there is at least Visi-Calc.)

Appallogy, continued...

* * *

This is, I suppose, also the place to respond to the September editorial in *Byte*. For those who missed it, or were mystified by it, it was an editorial *á clef* by Carl Helmers, the Founding Editor, in which several unnamed persons were accused of unethical conduct under vague and mysterious circumstances. If you didn't already know who or what it was about there was no way you could actually find out. However, it was all quite recognizable to a few hundred people, and indeed quite libelous.

The facts are these.

Scott Warren and Dennis Abbe, of Houston, Texas, have for two years been working on an actor-language interpreter for the Z80 under CP/M. In structure this most closely resembles Smalltalk, developed at Xerox PARC and discussed in our October and November issues.

In 1979 Warren and Abbe published a paper, referring to their language as an implementation of Smalltalk, and using the trade name Rosetta. They received one call from a member of the PARC team, who expressed interest in their work. Though their intentions were openly commercial, at no time did anyone from Xerox PARC (or elsewhere) correct or criticize their use of the name "Smalltalk."

Now precedent in this matter is plain. Languages are referred to generically, and anyone is free to create "a Basic" or "a Fortran" with any features he likes. (I have often said that you can sell any language as long as you *call* it Basic, and one compiler, "Basex," is widely mistaken for Basic even though the resemblance is debatable.) As the Rosetta effort resembled only one other language, Smalltalk, and as there had been no attempt to assert trademark on the name "Smalltalk," or correct or censure them for this usage, Warren and Abbe assumed that the use of this name had been in some sense "cleared." (And, indeed, no one could deny that theirs was in some important sense a *dialect* of Smalltalk.)

Okay, so I met these guys in March and thought their work was terrific; even though, obviously, it was far short of the Xerox Smalltalk, it would run on the Z80 and there was no guarantee we'd ever get anything else; The Überkorp might quash it.

So, as mentioned in earlier issues of this magazine, I made a couple of phone calls and got Warren and Abbe a place to show their wares at the National Computer Conference as guests of Exidy, Inc., running the language on a Sorcerer. I also helped with a couple of brochures: one explaining the general structure of the language, the other describing now I hoped to use it in conjunction with a long-term project of mine.

The names of Xerox PARC, Alan Kay et al., did not appear on the brochures, any more than the name of John Backus appears on Fortran brochures or the names of Kemeny, Kurtz and Dartmouth appear on advertisements for Basic. (However, in an article in the NCC issue of *Datamation* called "Introducing Rosetta Smalltalk," Warren and Abbe gave fulsome credit to the language's originators. Nor did I, in our symposium on actor languages, neglect to credit the PARC group.)

So much for the events which some people, at least the Founding Editor, seem to consider the crime of the century. It is alleged that in some way the Rosetta guys (and I, singled out for special obloquy as "the publicist" in the *Byte* thing) either tried to withhold credit from the creators of Smalltalk, hide its origin, or somehow pretend we had invented it ourselves.

This is almost too silly to reply to. But not quite.

The misunderstanding was a simple matter of paradigm. If you consider Smalltalk a generic language-name, like Fortran and Basic (and, since it has never been seen with a trademark sign, one might think that), then anyone can create a dialect and use that name in the time-hownored fashion. But if, as was made adundantly clear by certain people's reactions at the NCC, it is different from all other language-names in applying to one implementation only, then that could and should have been made clearer earlier. Such clarification could have been accomplished with much less hard feeling at such an earlier time.

Apparently people are still going around saying it was "obvious" that the term Smalltalk could *only* be applied to the Xerox product. Well, obvious is as obvious does; it does not hurt to state the obvious so that it will be obvious to others too. Those who insist that this misunderstanding could not have arisen in good faith force a diabolical and vicious construction upon these events, and are then required to squirrel around for a darker motive than commercialism and enthusiasm; they will not find it.

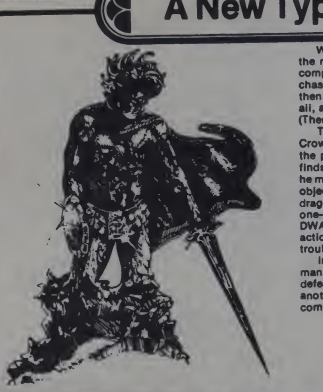
It all seems to be over now. The Rosetta people have agreed to change the name of their product, but aren't sure there's a market for what they've got.

The *real*, and indeed now the One and Only, Smalltalk is to be released (mirabile dictu!) sometime next year, so save your money for whatever it will run on. The software is free—and worth millions.

But in putting this matter away, I must also respond to the personal slurs by the Founding Editor on my own career. These were very clear: to wit, he alleged that I am simply an opportunistic and shallow flack looking for the Main Chance to become a parasite on someone else's serious endeavors. He put it surprisingly well, dismissing with unusual finesse my twenty years' work as designer and theorist of interactive computer systems. It is interesting to watch your work just swept aside like that, billowed away like a house on a flood. I have never seen the life-work of any living person dismissed in such a sweeping manner by a fellow professional, and it leaves me breathless. But I will fight back the temptation to reply in kind. And I will let pass the Founding Editor's statement that I and the Rosetta guys deliberately represented work by the people at Xerox PARC as our own. The statement is both libelous and asinine. Someone has indeed recently palmed off Xerox PARC work as his own—alas, in these very pages; but it was not, as mentioned earlier, I or Warren or Abbe.

In the spirit of general clarification, reconciliation and anticipation, I suggest we put all these matters to bed. We all need to get ready for Smalltalk, the One and Only. □

A New Type of Game



Welcome to an astonishing new experience! ADVENTURE is one of the most challenging and innovative games available for your personal computer. This is not the average computer game in which you shoot at, chase, or get chased by something, master the game within an hour, and then lose interest. In fact, it may take you more than an hour to score at all, and will probably take days or weeks of playing to get a good score. (There is a provision for saving a game in progress).

The original computer version of Adventure was written by Willie Crowther and Don Woods in Fortran on a PDP-10 at MIT. In this version the player starts near a small wellhouse. Upon entering the house, he finds food, water, a set of keys and a lamp. Armed with only these items, he must set out to explore the countryside in search of treasure and other objects of play. He must also confront dwarfs, snakes, trolls, bears, dragons, birds, and other creatures during his quest. The game accepts one-or two-word commands such as GET LAMP* SOUTH* or KILL DWARF. Of course, if you don't have the proper tool to carry out an action, or if you do something foolish, you may find yourself in big trouble.

In playing the game you wander thru various 'rooms' (locations), manipulating the objects there to try to find 'treasures'. You may have to defeat an exotic wild animal to get one treasure, or figure out how to get another treasure out of a quicksand bog. You communicate thru two-word commands such as 'go west', 'climb tree', 'throw axe', 'look around'.

Adventure

For Apple, TRS-80, Sorcerer, PET, CP/M

MISSION IMPOSSIBLE ADVENTURE (by Scott Adams) - Good Morning, Your mission is to... and so it starts. Will you be able to complete your mission in time? Or is the world's first automated nuclear reactor doomed? This one's well named, its hard, there is no magic but plenty of suspense. Good luck.....

THE COUNT (by Scott Adams) - You wake up in a large brass bed in a castle somewhere in Transylvania. Who are you, what are you doing here, and WHY did the postman deliver a bottle of blood? You'll love this Adventure, in fact, you might say it's LOVE AT FIRST BITE.....

ADVENTURELAND (by Scott Adams) - You wander through an enchanted world trying to recover the 13 lost treasures. You'll encounter WILD ANIMALS, MAGICAL BEINGS, and many other perils and puzzles. Can you rescue the BLUE OX from the quicksand? Or find your way out of the maze of pits? Happy Adventuring.....

VOODOO CASTLE (by Scott Adams) - Count Cristo has had a fiendish curse put on him by his enemies. There he lies, with you his only hope. Will you be able to rescue him or is he forever doomed? Beware the Voodoo Man.....

ORIGINAL ADVENTURE (by Crowther, Woods, Manning and Roichel) - Somewhere nearby is a collosal cave where others have found fortunes in treasures and gold, but some who have entered have never been seen again. You start at a small brick building which is the wellhouse for a large spring. You must try to find your way into the underground caverns where you'll meet a giant clam, nasty little dwarves, and much more. **This Adventure is Bi-Lingual**—you may play in either **English** or **French**—a language learning tool beyond comparison. Runs in 32K CP/M system (48K required for SAVE GAME feature). Even includes SAM76 language in which to run the game. The troll says "Good Luck."

PIRATE ADVENTURE (by Scott Adams) - "Yo Ho Ho and a bottle of rum..." You'll meet up with the pirate and his daffy bird along with many strange sights as you attempt to go from your London flat to Treasure Island. Can you recover LONG JOHN SILVER's lost treasures? Happy sailing matey.....

sensational software

TRS-80 Level II (16K) Machine language cassettes for only $14.95 each
CS-3007 Adventureland
CS-3008 Pirate Adventure
CS-3009 Mission Impossible
CS-3010 Voodoo Castle
CS-3011 The Count

TRS-80 Disk (32K) Menu-driven machine language disks for only $39.95 each
CS-3516 Adventureland, Pirate Adventure and Mission Impossible
CS-3517 Voodoo Castle, The Count and Ghost Town

Atari
CS-7003 Haunted House (16K Basic cassette, only $11.95)
CS-7502 Haunted House and 4 Outdoor Games (32K disk, $24.95)

Apple II (32K) Machine language cassettes for only $14.95 each
CS-4011 Adventureland
CS-4012 Pirate Adventure
CS-4013 Mission Impossible
CS-4014 Voodoo Castle
CS-4005 Haunted House (in Basic, only $11.95)

Apple II Disk (48K) Menu-driven machine language disks for only $39.95 each
CS-4513 Adventureland, Pirate Adventure and Mission Impossible
CS-4514 Voodoo Castle, The Count and Ghost Town
CS-4504 Haunted House and 4 Outdoor Games (only $24.95)

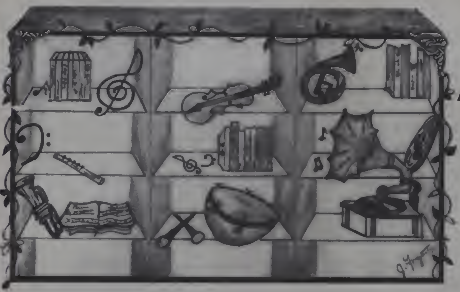
Pet (24K) Basic cassette
CS-1009 Adventureland and Pirate Adventure, only $19.95

Sorcerer (16K) Machine language cassettes for only $14.95 each
CS-5003 Adventureland
CS-5004 Pirate Adventure
CS-5005 Mission Impossible
CS-5006 Voodoo Castle
CS-5007 The Count

CP/M 8" Disk (48K) for only $24.95 each
CS-9003 Adventureland and Pirate Adventure (requires MBasic)
CS-9004 Original Adventure, bi-lingual with SAM76 language system.

Order directly from **Creative Computing Software Dept.** AFGG, P.O. Box 789-M, Morristown, NJ 07960. Send payment plus $1 shipping and handling. For faster service call in your bank card order to (800) 631-8112. In NJ call (201) 540-0445.

CIRCLE 300 ON READER SERVICE CARD



An Atari Library of Sound

Richard M. Kruse

Of the recognized human senses, it may easily be argued that the most important are those of sight and hearing. The movie industry was quick to realize the importance of adding sound to their visual productions. First there was simple background music, and later, when it became technically possible, sound was synchronized to the action. Few people today would pay to see a silent movie except under special circumstances.

Yet when most of us think of computers, we usually visualize someone sitting at a video console, typing, and staring silently into the screen. Hollywood generally adds some "bleeps" and "bloops", supposedly electronic, to the background. Real data processing centers are usually quite noisy with machinery running and several printers banging away. These are all artificial sounds, however, far removed from what all of us experience in daily life.

Personal computing, of course, need not follow the same path. If it is technically feasible, why not add the dimension of sound to the already accepted versatility of a good color graphics system? Why not, indeed! Manufacturers of small computers are responding in varying degrees to this challenge. It is now up to programmers to use this new capability effectively.

One of the outstanding features of the Atari 400/800 personal computers is the built-in sound generation system. There is no need to jury-rig an external amplifier and speaker and then operate it with "PEEK s" and "POKE s". Atari's sophisticated sound channels are manipulated through special Basic commands, and the RF output carries the sound information properly formatted to be reproduced through the speaker of a standard television receiver. The television's sound system does not have to be of especially high quality to adequately handle the range of frequencies produced (although it certainly doesn't hurt). An added bonus of this system is that sound and video are presented side-by-side. Most people will probably find this preferable to listening to a disembodied sound source physically separated from the visual presentation.

The Ataris give you not just a single sound generator, but four identical "channels" which may be used separately or in any combination. Each channel has individually controllable pitch and volume, along with a third parameter which Atari calls "tone." The Basic statement which activates one of the sound channels has the following form:

100 SOUND P1, P2, P3, P4

Parameters P1 through P4 are integer values. P1 specifies which channel is to be activated, identified as zero through three. P2 may be any value from 0 to 255, and sets the relative pitch or frequency of the sound. In the pure tone mode, the pitch range is about two and one-half octaves, and by using a look-up table of conversion factors between musical notes and pitch values, playing a melody on the Atari becomes almost trivial. Playing four-part harmony can be done with some additional programming effort.

It is now up to programmers to use this new capability effectively.

One of sixteen different volume levels (including off) is selected by the value of P4.

The tone parameter, P3, is a corker. There are eight possible values, two of which result in relatively pure musical tones. The remaining six, however, are not really "tones" at all, but special effects settings which produce strange and wonderful sounds that will be variously perceived as trucks, helicopters, heavy machinery, and warp drives. These effects, like the pure tones, may be varied in pitch and volume. And always, two or more sound channels may be active simultaneously. As you can see, the number of possible sounds and effects is staggering. Normal sounds can be imitated and new ones created, limited only by the imagination of the programmer.

To stimulate those imaginations, and to show the methods used to put these effects to work, one dozen varied and useful sound effects are presented here. Each effect is programmed as a subroutine which will run for a certain length of time and then terminate. Each subroutine makes use of one or more sound registers, and many of them accept one or more input parameters which modify the effect and/or its running time. A brief explanation is presented for each, so that you will be able to change the effects as desired.

1. Percussive Sound Generator-(See listing 1)

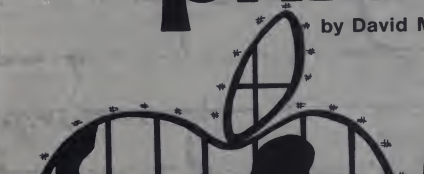
This is a "building block" subroutine which imitates the sound of struck or plucked musical instruments or, with different parameters, explosions or gunshots.

The percussive effect is achieved by executing a loop which initially sets a high volume level, then repeatedly reduces that level by a given percentage until it falls below a present minimum. The volume reduction factor is stored as the variable ICR, and it is easy to see that changing the value of ICR will change the rate of decay of the sound. Since ICR is calculated from the input parameter DUR, the decay rate can be modified at will each time the subroutine is called. The value 10 in

Richard M. Kruse. Xentrix Engineering. Box 8253. Wichita. KS 67220.

the Prisoner

by David Mullich



Inspired by the highly acclaimed television series, your Apple puts you in a nightmare 1984 world whose rulers seek to break you down by an extensive array of brainwashing techniques, while you are armed only with your intelligence and sense of individuality. Can you escape to freedom or will you remain forever THE PRISONER?

$29.95

"We want information . . ."

EDU-WARE T.M.

UNIQUE SOFTWARE FOR THE UNIQUE MIND.

Available at computer stores in finer villages everywhere. CA res. add 6% tx. Add $1.00 for shipping.

EDU-WARE SERVICES, INC • 22222 Sherman Way, #102 • Canoga Park, CA 91303
PH. # (213) 346-6783

CIRCLE 164 ON READER SERVICE CARD

statement 10020 is the tone parameter, and results in a pure tone output, so that this subroutine will imitate a chime or bell. Statement 10010 adds a brief burst of white noise at the start of the loop. (It is turned off at step 10025.) This enhances the initial "strike" effect and is heard in the sounds of many musical instruments. Statement 10040 turns the sound off altogether prior to returning to the calling program. While this percussive sound routine will run by itself, it can also be used in the generation of more complex sounds, as will be demonstrated.

2. Doorbell-(See listing 2)

...Now, who could that be?...

The familiar "Dinnng, donnng" of the doorbell is created by two sequential calls to a modified percussive routine. Two different pitches and two moderately long decays are used. What could be simpler?

3. Ringing Telephone-(See listing 3)

...Mildred, would you get that?...

The telephone bell is actually just repeated invocations of the percussive sound, using a high pitch and a short decay. Notice that the two sound registers are set at slightly different pitches. This creates the strident nature of this effect. The final percussive call uses a longer decay time, resulting in a fairly natural "lingering" sound. The apparently meaningless statement at line 10045 simply wastes some time between "rings." You will see this same type of delay in some of the other routines.

4. Alarm Bell-(Seeing listing 4)

...Attention all hands! Secure for hyperwarp...

This is another application of the percussive effect, and is almost identical to the telephone bell. The main differences are that this effect uses a lower pitch and a slower repetition rate. One subtle modification to the percussion routine in both of these effects is the use of a larger value in testing for the end of the decay (notice the variable LM). This is another way to modify the decay time and may be preferable for fast action.

5. Explosion-(Seeing listing 5)

...Hah! Got the little # @ * % !...

The explosion effect is also based on the percussive generator, using "white" (actually "pink") noise instead of a musical tone. For more volume we use three sound registers simultaneously, and to heighten the realism each is given a slightly different pitch. Finally, we use three different rates of decay, the slowest for the lowest pitch. This gives the "rolling" effect of a really "big bang." Entering this subroutine with DUR set to zero will give a pretty fair imitation of a gunshot, since it's basically the same kind of sound.

6. Siren #1-(See listing 6)

...Is he after me?...

This routine produces the rising and falling wail characteristic of electro-mechanical fire and police sirens. The inner loop in this subroutine (steps 10020 to 10035) generates either an increasing or decreasing pitch of constant amplitude. Each execution of the outer loop (steps 10015 to 10045) reverses the start, stop, and increment values. The delay is used again at step 10030 to waste a little time so that each execution of the loop takes about a second.

7. Siren #2-(See listing 7)

...Quickly, Henri! The Gendarmes...

This alternate siren effect, which I tend to think of as "European," is becoming more common in this country as well, as police and fire departments switch to purely electronic noisemakers. It is one of the simplest effects to create, requiring only alternating high and low pitches at constant volume. The wait loop is used again, at step 10025.

8. Ticking Clock-(See listing 8)

...You have ten seconds to guess the correct answer...

If you have been programming without sound, you will be amazed at the improvement to be gained by its use in games and audio-visual presentations.

The ticking of a clock (or bomb, heaven forbid) can be nicely simulated by repeated short bursts of white noise. Tone value eight, at a high pitch, serves this purpose. To get a tick-tock effect, two alternating values are used for the pitch parameter.

9. Klaxon-(See listing 9)

...RED ALERT! RED ALERT! Enemy sighted at...

Here, sound registers zero and one operate at slightly different pitches to generate a loud and strident blast, with sound register two filling in a buzzing effect. To add to the realism, one sound register is used at the beginning and end to build up to and decay from the main tone.

10. Whistle and Bomb-(See listing 10)

...Hit the deck!...

For this effect, the percussive explosion of example five is preceded by a convincing anticipatory whistle. Steps 10010 through 10030 create the whistle, which decreases in pitch while increasing in volume.

11. Steam Whistle-(See listing 11)

...All aboarrrrrd! Next Stop Pottsville...

A small amount of white noise from sound register zero in step 10025 adds a realistic hiss to this whistle variation. As in the Klaxon effect, there is a brief build-up preceding the main sound, and a decay at the end.

12. Sawing Wood-(See listing 12)

...And now for something completely different...

This final effect, unrelated to the others, is an example of picking a sound at random and trying to imitate it on the Atari. For sawing wood, you need a buzzing sound... Subroutine 10065. You need to make it rise and fall in pitch as the blade moves...subroutine 10030. For better realism, you need two different pitches as the blade is pushed forward on the cutting stroke and then returned... statements 10015 and 10020.

It is hoped that these relatively simple examples will provide the motivation for Atari owners to get the most out of one of the built-in features of their computers. Other possible effects might include animal imitations, automobile sounds, factory noises, and on and on...the list of possibilities is truly unbounded.

If you have been programming without sound, you will be amazed at the improvement to be gained by its use in games and audio-visual presentations. Once you grow accustomed to this added dimension, it is certain that you will no longer be satisfied with a dull, mute computer.

The secret to success of the small personal computer lies in your creativity and imagination. Put them to work with Atari sound and see what develops. You can't go wrong! □

LISTING 1: PERCUSSIVE SOUND GENERATOR

```
10000     REM PERCUSSIVE SOUND GEN
10002     REM ENTER W/2 PARAMETERS
10004     REM NTE=PITCH, 0-255
10006     REM DUR=LNGTH OF EFFECT, 0-10
10010     SOUND 1,5,8,6
10015     VOL=15: ICR=0.79+DUR/50
10020     SOUND 0,NTE,10,VOL
10025     SOUND 1,0,0,0
10030     VOL=VOL*ICR
10035     IF VOL>1 THEN 10020
10040     SOUND 0,0,0,0: RETURN
```

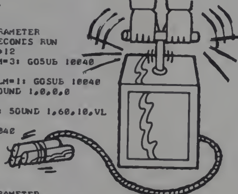
LISTING 2: DOORBELL

```
10000     REM DOORBELL
10002     REM NO ENTRY PARAMETERS
10010     NTE=105: DUR=7.5: GOSUB 10025
10015     NTE=132: DUR=8.5: GOSUB 10025
10020     SOUND 0,0,0,0: RETURN
10025     VOL=15: ICR=0.79+DUR/50
10030     SOUND 0,NTE,10,VOL
10035     VOL=VOL*ICR
10040     IF VOL>1 THEN 10030
10045     RETURN
```

LISTING 3: TELEPHONE BELL

```
10000 REM TELEPHONE BELL
10002 REM ONE ENTRY PARAMETER
10004 REM TMS=# RINGS
10010 FOR XX=1 TO 35
10015 IR=0.3: LM=2: GOSUB 10055
10020 NEXT XX
10025 IR=0.9: LM=1: GOSUB 10055
10030 SOUND 0,0,0,0: SOUND 1,0,0,0
10035 TMS=TMS-1
10040 IF TMS<1 THEN RETURN
10045 FOR WT=1 TO 300: NEXT WT
10050 GOTO 10010
10055 VL=15
10060 SOUND 0,40,10,VL
10065 SOUND 1,42,10,VL
10070 VL=VL*IR
10075 IF VL>LM THEN 10060
10080 RETURN
```

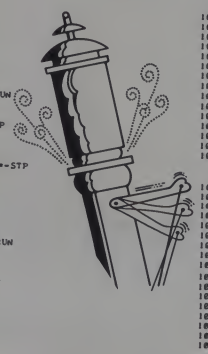
LISTING 4: ALARM BELL



```
10000 REM ALARM BELL
10002 REM ONE ENTRY PARAMETER
10004 REM DUR=APPROX SECONDS RUN
10010 FOR TMS=1 TO DUR*12
10015 VL=15: IR=0.5: LM=3: GOSUB 10040
10020 NEXT TMS
10025 VL=10: IR=0.95: LM=1: GOSUB 10040
10030 SOUND 0,0,0,0: SOUND 1,0,0,0
10035 RETURN
10040 SOUND 0,53,10,VL: SOUND 1,60,10,VL
10045 VL=VL*IR
10050 IF VL>LM THEN 10040
10060 RETURN
```

LISTING 5: EXPLOSION

```
10000 REM EXPLOSION
10002 REM ONE ENTRY PARAMETER
10004 REM DUR=LNGTH OF EFFECT, 0-10
10010 NTE=20: GOSUB 10025
10015 SOUND 1,0,0,0: SOUND 2,0,0,0
10020 RETURN
10025 SOUND 2,75,8,15
10030 ICR=0.79+DUR/100
10035 V1=15: V2=15: V3=15
10040 SOUND 0,NTE,8,V1
10045 SOUND 1,NTE+20,8,V2
10050 SOUND 2,NTE+50,8,V3
10055 V1=V1*ICR
10060 V2=V2*(ICR+0.05)
10065 V3=V3*(ICR+0.08)
10070 IF V3>1 THEN 10040
10075 SOUND 0,0,0,0: RETURN
```



LISTING 6: AMERICAN SIREN

```
10000 REM SIREN 1
10002 REM ONE ENTRY PARAMETER
10004 REM DUR=APPROX SECONDS RUN
10010 LO=50: HI=35: STP=-1
10015 FOR TIM=1 TO DUR
10020 FOR NTE=LO TO HI STEP STP
10025 SOUND 0,NTE,10,14
10030 FOR WT=1 TO 20: NEXT WT
10035 NEXT NTE
10040 XX=LO: LO=HI: HI=XX: STP=-STP
10045 NEXT TIM
10050 SOUND 0,0,0,0: RETURN
```

LISTING 7: EUROPEAN SIREN

```
10000 REM SIREN 2
10002 REM ONE ENTRY PARAMETER
10004 REM DUR=APPROX SECONDS RUN
10010 LO=57: HI=45: NT=HI
10015 FOR TIM=0 TO DUR*2
10020 SOUND 0,NT,10,14
10025 FOR WT=1 TO 180: NEXT WT
10030 NT=LO: LO=HI: HI=NT
10035 NEXT TIM
10040 SOUND 0,0,0,0: RETURN
```

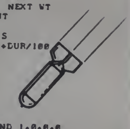
LISTING 8: TICKING CLOCK

```
10000 REM TICKING CLOCK
10002 REM TWO ENTRY PARAMETERS
10004 REM SIZ=1(FST) TO 10(SLW)
10006 REM DUR=APPROX SECONDS AT SIZ 5
10010 TIC=SIZ+5: TOC=SIZ+10
10015 FOR TIM=1 TO DUR
10020 SOUND 0,TIC,8,12: GOSUB 10035
10025 SOUND 0,TOC,8,12: GOSUB 10035
10030 NEXT TIM: RETURN
10035 SOUND 0,0,0,0
10040 FOR WT=1 TO SIZ*34: NEXT WT
10045 RETURN
```

LISTING 9: KLAXON

```
10000 REM KLAXON
10002 REM ONE ENTRY PARAMETER
10004 REM DUR=APPROX SECONDS RUN
10010 FOR TIM=1 TO DUR
10015 FOR NT=1 TO 10
10020 SOUND 0,100-NT,10,10
10025 NEXT NT
10030 SOUND 0,90,10,14
10035 SOUND 1,95,10,12
10040 SOUND 2,20,2,4
10045 FOR WT=1 TO 200: NEXT WT
10050 SOUND 1,0,0,0: SOUND 2,0,0,0
10055 FOR NT=1 TO 5
10060 SOUND 0,90+NT,10,8
10065 NEXT NT: SOUND 0,0,0,0
10070 FOR WT=1 TO 100: NEXT WT
10075 NEXT TIM: RETURN
```

LISTING 10: WHISTLE AND BOMB



```
10000 REM WHISTLE & BOMB
10002 REM ONE ENTRY PARAMETER
10004 REM DUR=LNGTH OF EFFECT
10010 V1=4: FOR NT=30 TO 75
10015 SOUND 0,NT,10,V1
10020 SOUND 1,NT+3,10,V1*0.7
10025 FOR WT=1 TO DUR*3: NEXT WT
10030 V1=V1*1.03: NEXT NT
10035 SOUND 2,35,8,12
10040 V1=15: V2=15: V3=15
10045 NT=DUR+5: ICR=0.79+DUR/100
10050 SOUND 0,NT,8,V1
10055 SOUND 1,NT+20,8,V2
10060 SOUND 2,NT+50,8,V3
10065 V1=V1*ICR
10070 V2=V2*(ICR+0.05)
10075 V3=V3*(ICR+0.08)
10080 IF V3>1 THEN 10050
10085 SOUND 0,0,0,0: SOUND 1,0,0,0
10090 SOUND 2,0,0,0: RETURN
```

LISTING 11: STEAM WHISTLE

```
10000 REM STEAM WHISTLE
10002 REM ONE ENTRY PARAMETER
10004 REM REM DUR=APPROX SECONDS RUN
10010 FOR VL=2 TO 14
10015 SOUND 1,56,10,VL: SOUND 2,66,10,VL
10020 NEXT VL
10025 SOUND 1,55,10,14: SOUND 0,5,8,3
10030 FOR WT=1 TO DUR*400: NEXT WT
10035 SOUND 0,0,0,0
10040 FOR VL=14 TO 1 STEP -2
10045 SOUND 1,55,10,VL: SOUND 2,67,10,VL
10050 NEXT VL
10055 FOR WT=1 TO 25: NEXT WT
10060 SOUND 1,0,0,0: SOUND 2,0,0,0
10065 RETURN
```

LISTING 12: SAWING WOOD

```
10000 REM SAWING WOOD
10002 REM ONE ENTRY PARAMETER
10004 REM DUR=APPROX SECONDS RUN
10010 FOR TMS=1 TO DUR
10015 ST=6: VL=12: GOSUB 10030
10020 ST=8: VL=8: GOSUB 10030
10025 NEXT TMS: RETURN
10030 FOR NT=ST+5 TO ST STEP -1
10035 GOSUB 10065: NEXT NT
10040 FOR NT=ST TO ST+5
10045 GOSUB 10065: NEXT NT
10050 SOUND 0,0,0,0: SOUND 1,0,0,0
10055 FOR WT=1 TO 25: NEXT WT
10060 RETURN
10065 SOUND 0,NT,2,VL
10070 SOUND 1,NT,8,VL*0.7
10075 WT=(WT/5)*5: RETURN
```

PRICE BREAKTHROUGH

We have used the VersaWriter to draw a picture of itself. Text may be added in any size or direction.

VersaWriter

High-Resolution Color Graphics for Apple II or Apple II Plus

The VersaWriter graphics tablet lets you create multicolor graphics and drawings with your Apple computer. It compares in quality to graphic bit pads and digitizers costing three times more money.

VersaWriter is a digitizer and software package which presents a new approach to hi-res graphics. It consists of a mylar plotting board with a clear plastic overlay. Attached to this board is the drawing arm, which has a magnifying lens with a cross-hairs at its end. You simply place any graph, picture or drawing (up to 8½" x 11") under the plastic overlay and "trace" it with the drawing arm. As you trace the drawing appears on the video screen.

The superior software of the VeraWriter enables you to do much more than just trace. Immediate commands include: color choice, brush size (the width of the drawing line), fill figure with color, draw a straight line between two points, use a different scale for drawing (.25 to 4), edit, erase, smothing factor (rounds off the rough edges as you draw), store picture on disk, and more.

One exceptional feature of the VersaWriter is the Shape Table function. You can take any picture, or portion of a picture, and store it as a shape table. Then the table can be recalled from memory and placed on any part of the screen. You can change the size of the image, rotate it, add to it, etc. By incorporating a series of images into a single shape table, commonly used symbols can be easily inserted into a variety of different programs. VersaWriter software includes an Electronic Drawing program which is a shape table of common schematic symbols-this program will give you a good idea of what the shape table can do, as well as let you easily produce electronic or logic diagrams.

Other programs included in the software are: the Textwriter, with which text can be added to graphics (UPPER & lower case, choice of color, text size, direction of text, starting point of text). Area/Distance-this program allows you to calculate distances (or perimeters) by establishing a measuring unit (of your choice) and tracing the shape or map route with the drawing arm. Areas of figures are calculated in the same way-this includes irregular and open figures. A very simple calibration program is also on this software disk.

A second software disk contains VersaWriter demonstration programs. For more advanced use of high-res graphics, there is a skeleton program which contains the guts of the VersaWriter. The VersaWriter is a sturdy peripheral device which plugs into the game paddles I/O port-the VersaWriter does not use up a card slot in the Apple computer. Also, the VersaWriter is not subject to the grounding problems and strong magnetic field problems of other, more expensive, hi-res graphic devices.

VersaWriter requires an Apple II with Applesoft in ROM (or an Apple II Plus), Disk, and a least 32K of memory.

VersaWriter comes complete with 8½" x 11" drawing surface, plastic overlay and two disks of software. Price $252.00 postpaid in continental USA. VersaWriter has a 90-day warranty on parts and labor.

Credit card customers include card number and expiration date of your Visa, Mastercard or American Express card. No C.O.D.'s. Bankcard customers may order toll-free to:

800-631-8112

(In NJ call 201-540-0445)

Dealer Inquiries Invited.

Peripherals Plus

119 Maple Ave., Morristown, NJ 07960

CIRCLE 239 ON READER SERVICE CARD

The Many Faces of Elementary Computer Graphics

Lynn Schwandt

```
GRCH

1  PRINT "                   ARXGROU                OUCHMARXGR"
2  PRINT "           CHOMARXGROUCHOM      UCHOMARXGROUCHOMA"
3  PRINT "       MARXGROUCHOMARXGROU   MARXOROUCHOMARXGROUCH"
4  PRINT "    MARXGROUCHOMARXGROUC   ARXOROUCHOMARXGROUCHOMARX"
5  PRINT " OROUCHOMARXGROUCHOMARX  OUCHOMARXGROUCHOMARXOROUCHOMA"
6  PRINT " HOMARXOROUCHOMARXORO   OMARXGROUCHOMARXOROUCHOMARXGROU"
7  PRINT "RXGROUCHOMARXGROUC      GROUCHOMARXOROUCHOMARXOROUCHOMARX"
8  PRINT "XGROUCHOMARXGRO            CHO   XGROUCHOMARXOROUCHOMARXGRO"
9  PRINT "UCHOMARXGRO                               ROUCHOMARXGROUCHOM"
10  PRINT " ARXGROUCH                                 OLMARXGROUCHOMARX"
11  PRINT " XOROUCHO                                   RXGROUCHOMARXGROU"
12  PRINT "  OUCHOM                                     ROUCHOMARXGROUCHO"
13  PRINT "    OMA                                       CHOMARXOROUCHOMAR"
14  PRINT "                                              XOROUCHOMARXOROUCH"
15  PRINT "                             ARXOROUCH        CHOMARXOROUCHOMARX"
16  PRINT "          HOMAR            GROUCHOMARXG       ARXOROUCHOMARXGROU"
17  PRINT "     HOMARXOROUCH        UCHOMARXGROUCHOM     OUCHOMARXGROUCHOMA"
18  PRINT " OMARXOROUCHMAARX       MARXGROUCHOMARXGROU   MARXOROUCHOMARXORO"
19  PRINT "XGROUCHOMARXGROUC       ROUCHOMARXOROUCHOMA    OUCHOMARXOROUCHOM"
20  PRINT " HO      OUCHO            ARXGR UCHOMARXO       ARXGROUCHOMARXG"
21  PRINT "RXO      MARXG  UC       RO     ARXGROU          UCHOMARXGROUCH  ARXG"
22  PRINT "U       OROUCHOMA       HO     OROUCHOMARX       ARXGROUCHOMAR   O  HOM"
23  PRINT "A      C  MAR   O       X     H         C         UCHOMARXORO   OM     ";
24  PRINT "RO"
25  PRINT "O     ARX     HOM       C      O        R          RXOROUCHO   XOROU   ";
26  PRINT "OM"
27  PRINT "MA      C    RX         R       O     ROU             CHOMARX   UCHOMARXO"
28  PRINT " OU          U          UC            HO     ROUCHO    OROUC  MARXOROUC"
29  PRINT "  AR        MA           RXO          RX     HO     OR      ARXOROUCHOMA"
30  PRINT "   UCH     GR               HOM     OU      X      OM      O  HOMARXOR"
31  PRINT "     XGROUCH                  ROUCHOM       C       RO     M RXGROUCH"
32  PRINT "           X                                RX       MA    ROUCHOMARXG"
33  PRINT "           C                  O             CHOM      OU   OMARXGRO  H"
34  PRINT "           RX                CHOMAR           R      ARXOROUCHO   XO"
35  PRINT "            CH       OUC    ARXGROUC         O         HOMARX     H"
36  PRINT "             XO      OMARXGROUCHOMARX        OR         XOROUCH    XO"
38  PRINT "        XOROUCHOMARXOROUCHOMARXOROUCH       OM         CHOM  X       ";
39  PRINT "HOM"
40  PRINT "        CHOMARXGROUCHOMARXOROUCHOMARX        RO         O   C       ";
41  PRINT "  R"
42  PRINT "        RXOROUCHOMARXOROUCHOMARXOROUC         M         H  AR       ";
43  PRINT "  O"
44  PRINT "        UCHOMARXGROUCHOMARXGROUCHOMA     OUCHOM R        X RO        ";
45  PRINT "  O"
46  PRINT "         RXOROUCHOMA   ROUCHOMARXGR       OM     C O       CHO       ";
47  PRINT " CH"
48  PRINT "                                     MARXOR    RXOR         XO       ";
49  PRINT " R"
50  PRINT "                      XOROUC        OR     OM      OM                ";
51  PRINT "OU"
52  PRINT "                    OUCHOM          CH      RO                        M"
54  PRINT "                       XORO         AR        MA                     OR"
56  PRINT "                          M        O  HOM                            H"
58  PRINT "                                   M    RO                          RX"
60  PRINT "                                   ROUCHOM                         OU"
62  PRINT "                                    M                              M"
64  PRINT "                                   GR                             OR"
66  PRINT "                                  CH                             CH"
68  PRINT "                                 AR                              R"
70  PRINT "                                RO                              OU"
72  PRINT "                               HO                               M"
74  PRINT "                            ARX                             RXGR"
76  PRINT "                          ORO                         RXGROU"
78  PRINT "                          H                         ROU"
80  PRINT "                          X                        HO"
82  PRINT "                          CH                      RX"
84  PRINT "                           XO                    OU"
86  PRINT "                            H                   OM"
88  PRINT "                            X                   O"
90  PRINT "                        GROUCHOMARXGROUCHOMARXOROUCHOMARXOR"
92  PRINT "                        HOMARXOROUCHOMARXOROUCHOMARXGROUCHO"
94  PRINT "                        XGROUCHOMARXOROUCHOMARXGROUCHOMARXG"
96  PRINT "                        CHOMARXGROUCHOMARXGROUCHOMARXOROUCH"
98  PRINT "                        RXOROUCHOMARXGROUCHOMARXOROUCHOMARX"
100  END

LEN
02391 WORDS = 10 RECORDS.  00518 RECORDS USED OF 00550 PERMITTED.
```

Lynn Schwandt, University of Northern Iowa, Cedar Falls, IA 50613.

```
GRO2

1   DIM S$[71],G$[83],L$[73]
2   S$="
3   FOR L=1 TO 64
4   R=INT(RND(0)*10)
5   G$="GROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCH"
6   L$[1,73]=G$[1+R,73+R]
7   READ B
8   READ E
9   IF E=0 THEN 14
10   L$[B,E]=S$[B,E]
11   IF L#49 THEN 7
12   L$[7,13]="GROUCHO"
13   GOTO 7
14   PRINT L$[1,B-1]
15   NEXT L
16   DATA 1,19,27,35,46,0,1,11,27,32,50,0,1,7,27,29,51,0,1,4,25,27,53,0
17   DATA 1,1,24,25,55,0,1,1,22,24,56,0,19,24,58,0,15,27,31,33,60,0,12,42
18   DATA 61,0,1,1,11,42,61,0,1,1,10,44,62,0,1,2,9,45,63,0,1,4,8,46,64,0
19   DATA 1,46,65,0,1,29,39,46,65,0,1,10,16,27,40,46,65,0,1,5,18,25,42
20   DATA 46,65,0,1,1,18,24,44,46,65,0,18,24,44,47,65,0,1,1,4,9,15,26,32
21   DATA 32,42,48,64,0,4,9,15,16,19,25,28,32,40,49,64,65,70,0,2,8,18,24
22   DATA 27,31,43,49,63,65,67,68,72,0,2,7,9,10,14,16,18,24,26,30,32,40,42
23   DATA 50,62,64,67,70,73,0,2,6,10,14,18,24,26,31,33,40,42,51,61,63,69
24   DATA 70,73,0,3,8,10,13,16,24,26,32,34,38,42,52,60,62,72,0,1,1,4,13,15
25   DATA 24,27,37,40,44,51,54,60,61,71,0,1,2,5,12,15,25,29,36,39,43,46,49
26   DATA 52,57,70,0,1,3,7,11,14,27,31,35,38,43,45,50,53,57,59,60,69,0
27   DATA 1,5,13,29,37,43,45,51,54,57,59,59,68,0,1,11,13,43,46,52,55,57,69,0
28   DATA 1,11,13,29,31,44,49,53,56,57,66,67,69,0,1,11,14,28,35,47,49,54
29   DATA 65,67,70,0,1,12,15,21,25,27,36,47,49,57,64,68,70,0,1,13,16,20
30   DATA 37,47,50,57,65,68,71,0,1,8,38,48,51,57,62,63,65,69,73,0,1,8,38,49
31   DATA 52,59,61,63,65,71,73,0,1,8,38,50
32   DATA 52,59,61,62,65,71,73,0,1,8,37,43,50,50,52,59,61,61,64,71,73,0
33   DATA 1,9,21,23,36,42,45,48,50,50,52,59,63,70,73,0,1,38,45,48,53,60
34   DATA 63,70,72,0,1,22,29,37,40,43,46,51,54,69,72,0,1,20,27,36,39,44
35   DATA 47,69,71,0,1,23,28,35,38,45,48,68,71,0,1,26,28,35,37,38,42,68
36   DATA 70,0,1,35,37,40,43,67,70,0,1,35,43,66,69,0,1,36,38,65,68,0
37   DATA 1,35,38,65,68,0,1,34,37,64,67,0,1,33,36,64,66,0,1,32,35,63,66,0
38   DATA 1,31,34,63,65,0,1,29,33,60,65,0,1,27,31,55,62,0,1,27,29,53,57,0
39   DATA 1,27,29,52,55,0,1,27,30,51,54,0,1,28,31,50,53,0,1,29,31,49,52,0
40   DATA 1,29,31,49,51,0,1,25,61,0,1,25,61,0,1,25,61,0,1,25,61,0,1,25,61,0
41   END

LEN
01975 WORDS = 08 RECORDS.  00518 RECORDS USED OF 00550 PERMITTED.
```

There is an abundance of programs that will produce computer pictures on the hard-copy terminal. Students at all levels of ability and accomplishment find some degree of fascination in obtaining copies of these programs so that they can demonstrate to less knowledgeable persons the power of the computer. Students at the Price Laboratory School are no different and computer produced "art work" is generated by nearly every student at one time or another.

One of the advantages of computer art is that computer novices often are drawn to their first computer experiences this way. Then, as the novelty wears off, they often develop a desire to learn about programming.

When students were informed that computer art would have to use less disk-space, they were faced with the option of producing more efficient programs or losing some of their favorite art programs. Lee Potter, a high-school senior in one semester programming course, accepted the challenge as a special project.

The programs demonstrate two step solution generated by Lee. GRCH, the first program, was the starting point.

Faces, continued...

GRCH is just a series of PRINT statements and consumes ten records of storage on the HP-2000. Programs of this type to produce ten different pictures would require one hundred records of disk storage. The problem was complicated by the desire to use the letters "GROUCHOMARX-GROUCHOMAR..." repeatedly to "sketch" the picture.

A big disadvantage is that the usual computer program for a picture consists of a number of print statements. This type of program requires little imagination, almost no programming skill and consumes a large amount of storage. Storage requirements are a particular problem when the shared-time system disk space is already utilized at 95% of capacity, and funds are unavailable for additional disk space.

The first partial solution, GRO2, involved using a character string of the repeated characters. G$="GROUCHOMARX-GROUCHOMARX...".A second character string S$ was loaded with all blanks. A third character string, L$, was built for each line to be printed. This was done by loading L$ with characters from G$ and then reading data to obtain the starting and ending points of the blanks that were required. The names G$, S$, L$ were chosen to represent GROUCH, SKIP and LINE respectively.

To provide variety, a random starting point was chosen in G$ before filling L$. Then beginning and ending coordinates for each blank space are read one pair at a time from the data. If the ending coordinate is zero, the final blanks are loaded into L$ and the line is printed. Otherwise another pair of coordinates is read and the blanks substituted in L$ until the ending coordinate is zero.

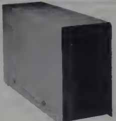
This first step was a decided improvement. GRO2 uses only eight records of storage as opposed to ten for GRCH. Lee was disappointed though. He had thought that the savings would be greater. The desire for a still more efficient storage approach led to the program GRO3 which uses

OR02

Buzzy

Hal Gerhardt

WE WILL NOT BE UNDERSOLD



DISK DRIVES $314

40 track. 102K Bytes. Includes power supply and TRS-80* compatible silver enclosure. Ready to plug-in and run the moment you receive it. Can be intermixed with each other and Radio Shack drive on same cable. 90 day warranty. One year on power supply. Available for 220 Vac (50 Hz) operation. **External card edge included.**

FOR TRS-80*		
CCI-100	5¼", 40 Track (102K Bytes) for Model I	$314
CCI-280	5¼", 80 Track (204K Bytes) for Model I	$429
CCI-800	8" Drive for Model II (½ Meg Bytes)	$795
For Zenith Z89		
CCI-189	5¼", 40 Track (102K Bytes) add-on drive	$394
Z-87	Dual 5¼" add-on drive system	$995

DISKETTES — Box of 10 (5¼") - with plastic library case

Maxell $30 **BASF** or **Verbatim** $24

8" double density for Model II (box of 10) $38

CLEAR PLASTIC CASE - Holds 50 diskettes $19

DISK OPERATING SYSTEMS

PATCHPAK #4 by Percom Data	$ 8.95
CP/M* for Model I, Zenith $145 • for Model II, Altos	$169.00
NEWDOS Plus 40track	$ 79.00
NEWDOS 80	$135.00

COMPLETE SYSTEMS

ALTOS 64K, DD, SS, 2-Drive, 1MB ACS 8000-2	$3395
APPLE 16K	$969
TRS-80* Model II-64K	$3499
TRS-80* Model III - 16K	$899
TRS-80* Expansion Interface	$249
ZENITH Z89, 48K all-in-one computer	$2440
ZENITH Z19	$735
TELEVIDEO 920C	$748
ATARI 400 $489 ATARI 800	$747
APF Game Only $ 95 Complete System	$489
MATTEL INTELLIVISION	$229

MONITORS

LEEDEX	12" B & W Video 100	$129
ZENITH	13" Color	$379
SANYO	9" B & W VM4509	$155
SANYO	12" B & W DM5012	$210
SANYO	12" Green Screen DM5112	$215
SANYO	13" Color DMC6013	$375
APF	9" B & W TVM-10	$139

TELECOMMUNICATIONS

LIVERMORE STAR MODEM 2-year guarantee	$145
CAT MODEM Works same as Radio Shack Telephone Interface II	$148
D-CAT HARD WIRED DIRECT MODEM	$189

COMMUNICATIONS SOFTWARE

CCI-TELNET VERSION 5: A communication package which enables microcomputer users to communicate both with large mainframes and other microcomputers. Completely CP/M compatible. Multiple communication protocols supported. $149

INTELLIGENT TERMINAL SYSTEM ST-80 III: Enables a TRS-80* to act as a dial-up terminal on any time sharing network. $139

16K MEMORY UPGRADE KITS 2 for $65 $35

200 ns for TRS-80*, Apple II, (specify): Jumpers $2.50

PRINTERS

NEC Spinwriter

Letter Quality High Speed Printer
Includes TRS-80* interface software, quick change print fonts, 55 cps, bidirectional, high resolution plotting, graphing, proportional spacing: R.O. $2395

R.O. with Tractor Feed $2575 KSR with Tractor Feed $2950

C.ITOH Starwriter, 25 CPS, daisy wheel printer	$1895
C.ITOH Starwriter II, 45 CPS, daisy wheel printer	$2195

Letter quality printers. Use up to 15" paper. 1 year warranty on parts. 3 months on labor. Proportional spacing and bidirectional printing. Same as VISTA V300.

779 CENTRONICS TRACTOR FEED PRINTER Same as Radio Shack line printer I	$969
737 CENTRONICS FRICTION & PIN FEED PRINTER n × 9 proportional and 7 × 8 mono spacing. Same as Radio Shack line printer IV	$795
730 CENTRONICS FRICTION & PIN FEED PRINTER 7 × 7 matrix Same as Radio Shack line printer II	$595
P1 CENTRONICS PRINTER Same as Radio Shack quick printer	$289
PAPER TIGER (IP445)	$899
(IP440) includes 2K buffer and graphics option	$879
(IP460) Bidirectional, 160 cps, graphics and 2K buffer	$1075
TI-810 Faster than Radio Shack line printer III. Parallel and serial w/TRS-80* interface software w/u + l case & paper tray	$1589
Compressed print, vertical form control	$1865
OKIDATA Microline 80 Friction and pin feed	$545
Tractor Feed, friction, and pin feed	$645
Microline 82 Bidirectional, friction and pin feed	$745
Microline 83 Bidirectional, 120 cps, uses up to 15" paper	$1050
EATON LRC 7000 + 64 columns, plain paper	$289
ANADEX DP-9500/01 $1350 DP-8000	$795

ACCESSORIES

HEAD CLEANING DISKETTE: Cleans drive Read/Write head in 30 seconds. Specify 5¼" or 8". **$20 ea/$45 for 3**

FLOPPY SAVER: Protection for center holes of 5¼" floppy disks. Installation tools and rings for 25 diskettes. $ 11.95
Re-orders of rings only $ 8.95

EXTERNAL DATA SEPARATOR: Eliminates data separation problems (crc). Improves reliability. This plug in unit comes fully assembled and tested. $ 29.95

Z-80 SOFTCARD: Your key to software expansion. The plug-in Z-80 Softcard transforms your Apple into a Z-80 while keeping all the benefits of the 6502. Comes with CP/M in two disk format, MBASIC and GBASIC, full documentation and utility programs. $339

RF MODULATOR: Adapts video to TV	$ 35.00
TRS-80 & OTHER MYSTERIES	$ 18.95
NEC SPINWRITER THIMBLE $11.95 **RIBBON**	$ 8.00
CCS CARDS: Parallel or serial printer interface cards	$115.00
RS232: For Radio Shack Interface.	$ 84.00
TRS232: Teletype current loop output from cassette port	$ 49.00
DISK-DRIVE EXTENDER CABLES: Fits all mini-disk drives.	$ 16.95
SIX (6) PRONG ISOLATOR: ISO-2	$ 54.00
AC FILTER/6 PRONG POWER STRIP	$ 39.00
DISK DRIVE CABLES: 2 drive $29.00 4 drive	$ 35.00
DUST COVERS: TRS-80/Apple	$ 7.95
PLASTIC DISKETTE HOLDER: For ring binder, holds 20	$ 8.00

For fast delivery, send certified checks, money orders or call to arrange direct bank wire transfers. Personal or company checks require two to three weeks to clear.

DEALER (NATIONAL/INTERNATIONAL) INQUIRIES INVITED **Send for FREE Catalogue**

The CPU SHOP

5 Dexter Row, Dept. CCOIM
Charlestown, Massachusetts 02129
Hours 10AM-6PM (EST) Mon.-Fri. (Sat. till 5)

TO ORDER CALL TOLL FREE 1-800-343-6522
TWX: 710-348-1796 Massachusetts Residents call 617/242-3361

Massachusetts Residents add 5% Sales Tax
*TRS-80 is a Tandy Corporation Trademark
• Digital Research

master charge VISA

CIRCLE 171 ON READER SERVICE CARD

Faces, continued...

"packed" data to store the information about where the spaces should be.

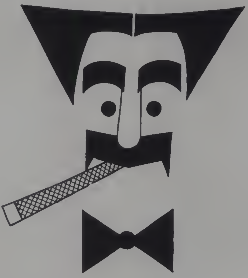
Note that the data of line 16 in GRO2 contains 1,19,27,35,46 and 0 for data for the first ouput line. GRO2 uses these six data items to generate and print the first line. Observe the data line 37 in GRO3 contains 192735., 460000.,... GRO3 uses these two data items to generate and print the same first line of the picture. The major difference being that GRO3 contains a section of program statements that "unpack" the data items and generate the appropriate blank spaces in the line to be printed.

While GRO3 requires only few records of storage, there is probably a more efficient approach than the one illustrated in GRO3. The purpose was not to develop the best approach, but to develop a better program than the previous one and still have results which are understandable by average high school students. The reader is invited to generate and share other approaches which are more efficient but yet understandable. □

The purpose was not to develop the best approach, but to develop a better program than the previous one.

GRO3

```
1   DIM S$[71],G$[83],L$[73],A[20]
2   S$=' 
3   FOR L=1 TO 65
4   R=INT(RND(0)*10)
5   G$='GROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCHOMARXGROUCH'
6   L$[1,73]=G$[1+R,73+R]
7   X=1
8   READ N
9   IF N=0 THEN 22
10   IF N<0 THEN 24
11   IF X#1 THEN 13
12   A[1]=1
13   D=10000
14   FOR U=1 TO 3
15   X=X+1
16   A[X]=INT(N/D)
17   IF A[X]=0 THEN 27
18   N=N-INT(N/D)*D
19   D=D/100
20   NEXT U
21   GOTO 8
22   A[X+1]=0
23   GOTO 27
24   X=0
25   N=-N
26   GOTO 13
27   FOR I=1 TO X STEP 2
28   B=A[I]
29   E=A[I+1]
30   IF E=0 THEN 35
31   L$[B,E]=S$[B,E]
32   IF L#49 THEN 34
33   L$[7,13]='GROUCHO'
34   NEXT I
35   PRINT L$[1,B]
36   NEXT L
37   DATA 192735.,460000.,112732.,500000.,72729.,510000.,42527.,530000.,12425
38   DATA 550000.,12224,560000.,-192458.,0,-152731.,335800.,-124259.,0,11142
39   DATA 600000.,11044,610000.,20945,610000.,40846.,630000.,466400.,293946.
40   DATA 640000.,101627.,404664.,0,51825.,424664.,0,11824,444664.,0,-182444.
41   DATA 476400.,10409,152632.,324248.,630000.,-40915.,161925.,283240.,496465.
42   DATA 690000.,-20818,242731.,434963.,656769.,710000.,-20709,101416.,182426.
43   DATA 303240.,425062.,646770.,720000.,-20610,141824.,263133.,404251.,616368.
44   DATA 717200.,-30810,131624.,263234.,384252.,606271.,0,10413,152427.,374044.
45   DATA 515460.,617000.,20512,152529.,363943.,464952.,576900.,30711,142731.
46   DATA 353843.,455053.,575960.,680000.,51329.,374345.,515457.,595967.,0,111343.
47   DATA 465255.,576800.,111329.,314449.,535657.,666768.,0,111428.,354749.
48   DATA 546567.,690000.,121521.,252736.,474957.,646869.,0,131620.,374750.
49   DATA 576568.,700000.,83848.,515762.,636569.,720000.,83849.,525961.
50   DATA 636571.,720000.,83850.,525961.,626571.,720000.,83743.
51   DATA 505052.,596161.,647172.,0,92123.,364245.,485050.,525963.,707200.,384548.
52   DATA 536063.,707100.,222937.,404346.,515469.,710000.,202736.,394447.,697000.
53   DATA 232835.,384548.,687000.,262835.,373842.,686900.,353740.,436769.,0
54   DATA 354366.,680000.,363865.,670000.,353865.,670000.,343764.,660000.
55   DATA 333664.,650000.,323563.,650000.,313463.,640000.,293360.,640000.
56   DATA 273155.,610000.,272953.,560000.,272952.,540000.,273051.,530000.
57   DATA 283150.,520000.,293149.,510000.,293149.,500000.,256000.,256000.
58   DATA 256000.,256000.,256000.
59   END
LEN
00990 WORDS = 04 RECORDS.  00518 RECORDS USED OF 00550 PERMITTED.
```



MODEL II

26-4002
64K 1 Drive
$3440.00

26-4160 1 Drive EXP	$1035.00
26-4161 2 Drive EXP	1575.00
26-4162 3 Drive EXP	2115.00
26-4501 Gen. Ledger	180.00
26-4502 Inventory	180.00
26-4503 Payroll	360.00
26-4554 Acct. Rec.	180.00
26-4701 Fortron	270.00
26-1157A Daisy Wheel	2495.00
26-1158 Daisy Wheel II	1799.00

$ DISCOUNT $
TRS-80® DEALER

COMPUTER SPECIALISTS

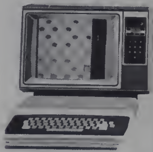
26-1155 Quick Printer II	$187.00
26-1145 RS-232 Board	84.00
26-1140 "O" K Interface	249.00
26-1141 "16" K Interface	359.00
26-1142 "32" K Interface	469.00
26-1160 Mini Disk - Drive O	419.00
26-1161 Mini Disk - Additional	419.00
26-1154 Lineprinter II	699.00
26-1156 Lineprinter III	1799.00
26-1159 Lineprinter IV	859.00
26-1166 Line Printer VI	1080.00
26-1563 Scripsit - Disk	79.00
26-1566 Visicalc	83.00
26-1562 Profile	72.00

NOTE: Call for availability of VIDEO TEX, Model III, Color, and other new products.

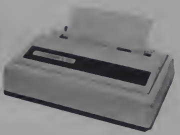
MODEL III

26-1061 4K I	$630.00
26-1062 16K III	888.00
26-1063 32K III 2-Drives, RS232	2225.00

COLOR



26-3001 4K	$360.00
26-3002 16K	540.00
26-3010 Color Video	360.00
26-1206 Recorder	54.00
26-3008 Joysticks	22.50



CENTRONICS

Fast 100 CPS Centronics
730 Printer.................. $659.00
Text Quality Centronics
737 Printer.................. $819.00

Model II Cobol Compiler
$360.00
Cobol Run Time Package
$36.00

ALL OTHER R.S. SOFTWARE FURNITURE, STANDS, CABLES AND ACCESSORIES AT DISCOUNT FROM CATALOG PRICE.

Novation Cat Modem..$149.00
CCA Data Management System..............72.00
Adventure Games
Games 1-9 each.........14.00

Pocket Computer

26-3501 1.9K P.C.	$225.00
26-3503 Cassette I/F	45.00
14-812 Recorder	72.00

Acorn
Software
Products, Inc.

GAMES:

Alien Invasion	$9.00
Stock Market	9.00
Star Trek	9.00
Block 'Em	9.00
Ting-Tong	9.00

UTILITIES:

System Savers	14.00

EDUCATION:

Language Teacher	18.00

FREE: COMPUTER CATALOG UPON REQUEST

1-800-841-0860 Toll Free Order Entry

MICRO MANAGEMENT SYSTEMS, INC.

No Taxes on Out Of State Shipments

Immediate Shipment From Stock on Most Items

DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
(912) 377-7120 Ga. Phone No.

*TRS-80 is a registered trademark of the Tandy Corp.

R.S. 90 Day Limited Warranty
F-48 Form Provided

Largest Inventory
In the S.E. U.S.A.

CIRCLE 183 ON READER SERVICE CARD

Hexagons and Ellipses the Easy Way

Gordon Speer

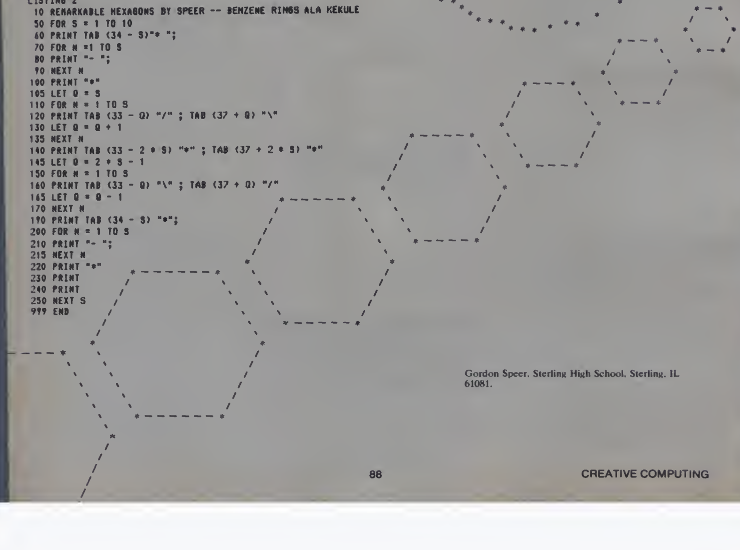
Here is proof, twice over, that graphics programs can be short, simple, and elegant. Listing #1 produces ellipses with any chosen major axis and any minor axis that has a value less than seven. The second listing (both programs are by Gordon Speer) produces a series of hexagons, each one larger than the last. Using the same approach, readers might wish to try producing other figures that can be printed in this manner. — DL

LISTING 1

```
10 DIM M(2)
20 INPUT M(1),M(2)
30 LET A = 25 * (M(2) ^ 2)
40 LET B = 3 * (M(1))
50 FOR Y = B TO -B STEP -1
60 LET X = INT (SQR (ABS (A - (A * Y ^ 2) / B ^ 2 )))
70 PRINT TAB (35 - X) "*" ; TAB (35 + X) "*"
80 NEXT Y
90 END
```

LISTING 2

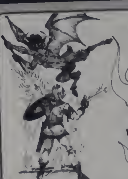
```
 10 REMARKABLE HEXAGONS BY SPEER -- BENZENE RINGS ALA KEKULE
 50 FOR S = 1 TO 10
 60 PRINT TAB (34 - S)"* ";
 70 FOR N =1 TO S
 80 PRINT "- ";
 90 NEXT N
100 PRINT "*"
105 LET Q = S
110 FOR N = 1 TO S
120 PRINT TAB (33 - Q) "/" ; TAB (37 + Q) "\"
130 LET Q = Q + 1
135 NEXT N
140 PRINT TAB (33 - 2 * S) "*" ; TAB (37 + 2 * S) "*"
145 LET Q = 2 * S - 1
150 FOR N = 1 TO S
160 PRINT TAB (33 - Q) "\" ; TAB (37 + Q) "/"
165 LET Q = Q - 1
170 NEXT N
190 PRINT TAB (34 - S) "*";
200 FOR N = 1 TO S
210 PRINT "- ";
215 NEXT N
220 PRINT "*"
230 PRINT
240 PRINT
250 NEXT S
999 END
```



Gordon Speer, Sterling High School, Sterling, IL 61081.

THE PROGRAM STORE

CALL TOLL FREE 800 424-2738



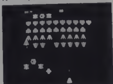
HELLFIRE WARRIOR

From Automated Simulations
A sequel to the famous "Temple of Apshai", HELLFIRE WARRIOR is expanded to include more command options, more potions, more magical items and more special effects. In addition to an Innkeeper, HELLFIRE WARRIOR has an armorer, apothecary and magic shop.

WARNING: not for beginning Dunjonquest players! If you are new to these fascinating games, start out with "Temple of Apshai", "Morloc's Tower" or "Datestones of Ryn".

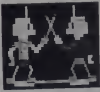
	Tape	Disk
HELLFIRE WARRIOR	$24.95	$29.95
TEMPLE OF APSHAI	$24.95	$29.95
MORLOC'S TOWER	$14.95	$19.95
DATESTONES OF RYN	$14.95	$19.95

GALAXY INVASION



By Bill Hague from Big-Five
"The rage of the arcades" is now available for the TRS-80! Exciting sound effects add to the action as the invaders swoop down to destroy your base. Even while you have your hands full battling the aliens, you have to watch out for the Flagship! Super graphics, super action, super fun!

Level I or II, tape...$14.95

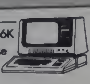


DUEL «N» DROIDS

By Leo Christopherson from Acorn
Your 'droid has already learned NIM, so now it's time to teach it how to wield a laser sword! Leo Christopherson, author of "Android NIM," "Dancing Demon" and other animations, has developed a new type of animation and high-quality sound in his latest work.

Your 'droid starts out as a lowly clown. You teach it how to use a laser sword by controlling its movements. After training it to be a "Grand Master," you enter the tournament against the program's skilled 'droid! Entertainment for all ages.

Protected Tape...$14.95
Protected Disk...$20.95



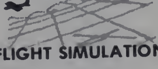
TRS-80 Level II 16K unless otherwise noted

WARGAMES

From Computer Simulations
For the serious war gamer, these three World War II strategic simulations offer incredible historical accuracy. Take command of the U.S. and Allied armies, try various approaches to specific WWII bettles. Would you have faired better than the actual result - or worse?
Each war game displays a combat map, with notation of the topography, towns, enemy positions and type. You move your units, manage your supplies, and carry on the fighting day-by-day. For real challenge, choose from these three games. Available on tape.

BATTLE OF THE BULGE: St. Vith....$14.95
BATTLE OF THE BULGE: Bastogne...$19.95
D-DAY: The Invasion of France......$19.95

Unbelievable Realtime 3-D Graphics!



FLIGHT SIMULATION

From Sub-Logic

The wait is over! If 3-D graphics seem impossible on the low resolution TRS-80, you haven't seen this brilliant program. During FLIGHT SIMULATION, you instantly select instrument flight, radar, or a breathtaking pilot's-eye-view. But be sure to strap yourself in you're liable to get dizzy!

Once you put in some air time learning to fly your TRS-80, head for enemy territory and try to bomb the fuel depot and airstrip while fighting off five enemy warplanes. Good Luck!

Level I or II Tape...$25.00

INVADERS FROM SPACE

By Carl Miller from Acorn
A new and faster machine language approach to this classic (and eddictive) space game. As you play, the aliens are dropping bombs, moving from side to side, and trying to overrun your bases. You try to by shooting at them, and your score grows larger with each hit. But, just as you think you've got the invaders under control, they speed up their action.

INVADERS FROM SPACE offers variable game speed, enemy bomb frequency and accuracy, number of shots on screen and number of your bases. Move your bese and simultaneously fire at the invaders which you cannot do in most other similar games. Full sound effects, incredible speed and action!

Protacted Tape...$14.95
Protected Disk...$20.95

ELECTRONIC BASKETBALL HANDICAPPER

By Sothen, Laurence & Gavenda from Acorn
Basketball is the first of the Electronic Handicapper Series from Acorn. It will introduce you to the benefits of predicting the winners of this season's basketball games. This two-tape package gives you power ratings to get you started, then you keep the the data tape informed of game scores as they are played. The program will calculate a projected winner and point spread, based on available data. Last season Handicapper was able to predict 85% of the winners, with 64% point spread accuracy. Requires only 16K.

On Tape...$99.00

Z80ZAP

From Org-Tex
New machine language disk access/modification program. With Z80ZAP, you will be able to...
...Reed, Display and Modify disk sectors ...Remove Passwords from any disk file ...Zero Out any disk sector ...Recover killed or lost disk files ...Compare disk sectors, byte-by-byte ...Find any designated byte in a disk sector ...Apply patches, fixes, etc.

Z80ZAP automatically calculates the Hash index Code for any file and tells you exactly how to use it when recovering killed files. The flashing cursor acts as a pointer to the byte on which you are working plus its ASCII equivalent, making diract disk editing considerably easier. Designed to outperform SUPERZAP in speed and capabilities, Z80ZAP is a "must" for disk drive owners.

On Disk.....$29.95

BOSS

By V. Hester from Soft Sector
This utility is the perfect tool for creating and debugging Basic programs. It allows single stepping through the Basic program, setting up to five breakpoints within the program and tracing of program logic using only a small portion of the display screen. With BOSS, you can review selected variables during program exacution and return to the program with the display restored. Allows storing programs in high memory for later retrieval. For Level II, TRSDOS, NEWDOS+, NEWDOS/80, VTOS.

On Tape...$29.95

DISASSEMBLER

By Roy Soltoff from Misosys & Acorn
A two-pass disassembler for TRS-80 that converts machine code to Z-80 assembly language listings. DISASSEMBLER produces symbolic labels with output to video, printer or tape. Radio Shack's Editor/Assembler will read and load the tapes for easy modification and reassembly. Extend the capabilities of Editor/Assembler with this utility. On tape for two different memory locations.

$14.95

Visit Our New Store: W. Bell Plaza – 6600 Security Blvd · Baltimore, MD

TO ORDER CALL TOLL FREE 800 424-2738 For information Call (202) 337-4691

MAIL ORDERS: Send check or M.O. for total purchase price, plus $1.00 postage & handling. D.C. residents, add 5% tax. Charge card customers: include all ambossed information on card.

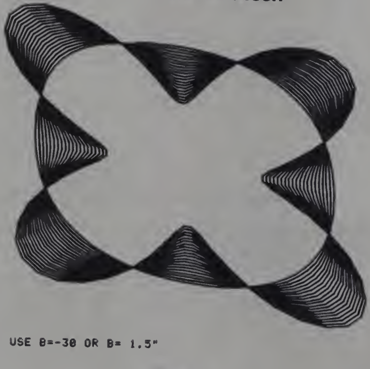
THE PROGRAM STORE
4200 Wisconsin Avenue NW, Dept. R7 Box 9609
Washington, D.C. 20016

CIRCLE 175 ON READER SERVICE CARD

Grapefruit Delight*

*and other surprises

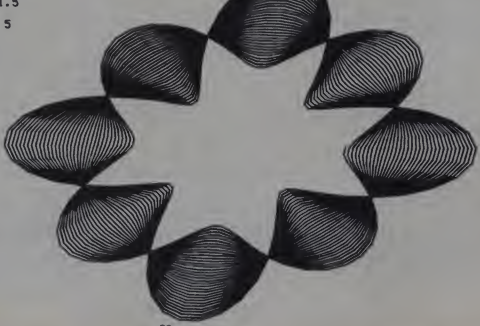
Joe Jacobson



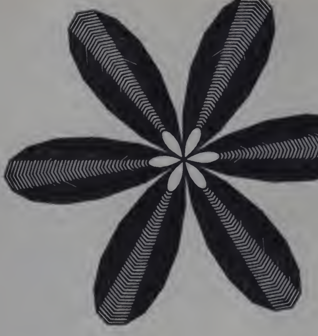
The programs shown below, written for a Tektronix 4051, demonstrate an interesting approach to graphics. Each program produces a specific type of figure, but the results can vary widely depending on the input. As you can see, this method produces pleasing and surprising results. □

Joe Jacobson, 1602 Upland Ave., Jenkintown, PA 19046.

```
100 REM RING SINE
101 PRINT "B IS THE INITIAL AMPLITUDE; USE B=-30 OR B= 1.5"
102 PRINT "ENTER B"
103 INPUT B
105 SET DEGREES
106 WINDOW -100,100,-100,100
107 VIEWPORT 0,130,0,100
110 PAGE
117 FOR A=B TO 30 STEP 1.5
119 MOVE 60,0
120 FOR T=0 TO 360 STEP 5
130 R=60+A*SIN(4*T)
135 X=R*COS(T)
136 Y=R*SIN(T)
140 DRAW X,Y
150 NEXT T
160 IF A=>30 THEN 170
165 NEXT A
170 END
```

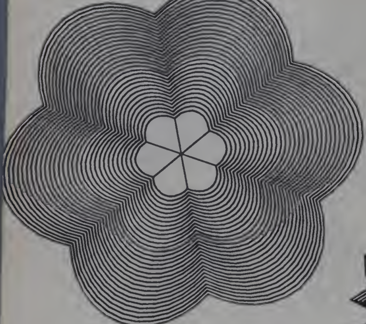


```
89 REM THREE-LEAF CLOVER
100 PAGE
109 SET DEGREES
110 WINDOW -500.500,-500.500
111 VIEWPORT 15.115.0.100
112 PAGE
113 PRINT "ENTER L"
114 INPUT L
115 PAGE
120 FOR B=100 TO 500 STEP 10
130 FOR A=0 TO 360 STEP 5
140 GOSUB 180
145 IF A>0 THEN 150
146 MOVE X.Y
147 GO TO 160
150 DRAW X,Y
160 NEXT A
170 NEXT B
180 R=B*ABS(SIN(L*A))
190 X=R*COS(A)
200 Y=R*SIN(A)
210 RETURN
220 END
```

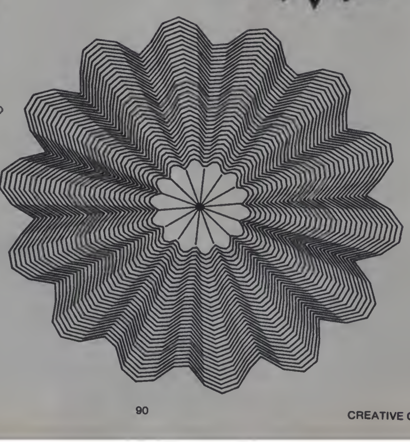


```
89 REM GRAPEFRUIT DELIGHT
90 DIM R1(3)
100 PAGE
109 SET DEGREES
110 WINDOW -501.501.-501,501
111 VIEWPORT 15,115.0,100
112 PAGE
113 L=6
120 B=300
130 FOR A=0 TO 360 STEP 5
140 GOSUB 180
145 IF A>0 THEN 150
146 MOVE X,Y
147 GO TO 160
150 DRAW X.Y
160 NEXT A
175 GO TO 219
180 R=B*(1+0.1*ABS(SIN(L*A)))
190 X=R*COS(A)
200 Y=R*SIN(A)
210 RETURN
219 MOVE B/3.0
220 FOR A=0 TO 360 STEP 5
230 R=B/3*(1+0.1*ABS(SIN(1.5*A)))
240 X=R*COS(A)
250 Y=R*SIN(A)
255 DRAW X,Y
260 NEXT A
270 C=1.1*B+15
280 MOVE C,0
290 FOP A=0 TO 360 STEP 5
300 X=C*COS(A)
310 Y=C*SIN(A)
320 DRAW X,Y
330 NEXT A
340 R1(1)=B/3*(1+0.1*ABS(SIN(1.5*60)))
350 R1(2)=B/3*(1+0.1*ABS(SIN(1.5*30)))
360 R1(3)=B/3
370 A1=0
380 J=3
390 GOSUB 500
400 A1=60
410 J=1
420 GOSUB 500
430 A1=30
440 J=2
450 GOSUB 500
460 A1=90
470 J=2
480 GOSUB 500
490 GO TO 600
500 FOR M=1 TO 3 STEP 1
510 A=A1+(M-1)*120
520 X=B*COS(A)
530 Y=B*SIN(A)
540 MOVE X,Y
550 X=R1(J)*COS(A)
560 Y=R1(J)*SIN(A)
570 DRAW X,Y
580 NEXT M
590 RETUPN
600 WINDOW 0.180,-2.01,3.01
610 VIEWPORT 61,69,46,54
620 D=-2
630 GOSUB 690
640 D=0
650 GOSUB 690
660 D=2
670 GOSUB 690
680 GO TO 740
690 MOVE 0,D
695 FOR X=0 TO 180 STEP 2
700 Y=SIN(X)+D
710 DRAW X.Y
720 NEXT X
730 RETURN
740 END
```

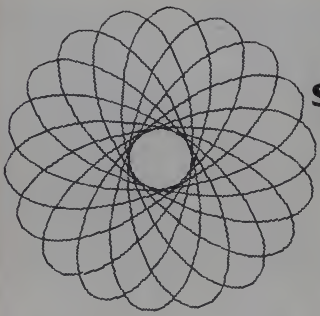
GRAPEFRUIT DELIGHT



```
09 REM FLUTED SCALLOPS
100 PAGE
109 SET DEGREES
110 WINDOW -501,501,-501,501
111 VIEWPORT 15,115,0,100
112 PAGE
113 PRINT "ENTER L"
114 INPUT L
115 PAGE
120 FOR B=100 TO 400 STEP 10
130 FOR A=0 TO 360 STEP 5
140 GOSUB 180
145 IF A>0 THEN 150
146 MOVE X,Y
147 GO TO 160
150 DRAW X,Y
160 NEXT A
170 NEXT B
175 GO TO 211
180 R=B*(1+0.25*ABS(SIN(L*A)))
190 X=R*COS(A)
200 Y=R*SIN(A)
210 RETURN
211 FOR N=0 TO L-1 STEP 1
212 R=100
213 T=N*(180/L)
214 X=R*COS(T)
215 Y=R*SIN(T)
216 MOVE X,Y
217 X=R*COS(T+180)
218 Y=R*SIN(T+180)
219 DRAW X,Y
220 NEXT N
221 END
```




AIR TRAFFIC
CONTROLLER
In Air Traffic Controller you assume responsibility for the safe flow of air traffic within a 15x25 mile area up to 5,000 feet in altitude. During your shift as a controller in charge of this airspace, 26 aircraft become active and under your control. Jets and prop planes have to be guided to and from the two airports, navigational beacons and ten entry/exit fixes. The aircraft enter the controller's airspace at various altitudes and headings whether or not you are ready.
Air Traffic Controller retains the basic realism of air traffic control. This program requires the same steady nerves under pressure and the same instant, almost instinctive, analyses of complex emergencies which are demanded of a professional air traffic controller. But "ATC" adds the excitement and well-defined goals of a game. This is just a simulation, and all passengers left in air traffic limbo by a panicked player will live to fly another day.
Your goal is to get all of the aircraft to their assigned destination before the shift is completed. At your disposal are a radar display of the aircraft positions in the control area; coded information concerning aircraft heading, destination and fuel supply; navaids enabling you to hold aircraft or assign them automatic approaches; and commands to alter the altitude and heading of the aircraft. Working against you are altitude and heading requirements, fuel restrictions and, of course, the inimitable clock.
The most obvious measure of difficulty of a game is the clock setting at the beginning. In a 99 minute game you will have time to go fix a sandwich between the appearance of two successive aircraft, while in the 16 minute game you may not have time to swallow before all of the aircraft have appeared.
No two games, even at the same clock setting, are ever alike. As controller, you must cope with the unique requirements of each aircraft. The game will end if you commit a "boundry error," that is, if an aircraft fails to leave your area at the proper altitude and exit fix...causing an unpleasant surprise for the controller next door. The game also ends if you fail to leave a comfortable margin of safety between the aircraft as they whiz past each other. In cases of excessive delay, fuel supply considerations will become invested with a particular sense of urgency.
Successful guidance of all aircraft to their destination is a heady accomplishment. This never fails to thrill ATC enthusiasts at each successive level of play.
Your local retail store should carry Creative Computing Software. If your favorite retailer does not carry the software you need, have him call in your order to (800) 631-8112. Or you can order directly from Creative Computing. Write to Creative Computing Software, P.O. Box 789-M, Morristown, NJ 07960. Air Traffic Controller is now available for the 16K TRS-80 (3006), for the 16K Apple II and Apple II Plus (4008), the 8K Sorcerer (5008) and for the 4K Sol-20 (8001). All are on cassette for $9.95. Include $1.00 for postage and handling. For faster service, call in your bank card order toll free on our order hotline, (800) 631-8112.
Prices are subject to change without notice.
creative computing software
sensational software



Spiro-Graph

Brian Sietz

Introduction

When I was younger, I remember making pretty designs with my spiro-graph toy. To make a design, I would take two wheels with gears around the outside and tack one into a piece of paper. The other wheel would either rotate outside or inside of the first wheel. The pen would be in a hole on the rotating wheel, and would leave a trail behind it as the wheel turned.

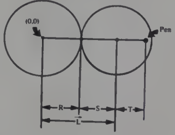
This program is designed around the same principle. There are four pieces of data which this program needs to run. These are R, S, T and Sign. R is the radius of the stationary circle, S is the radius of the moving circle, and T is the distance from the center of circle S to where the pen is placed. Sign will be either-1, meaning circle S is inside circle. R, or 1, meaning that circle S is outside circle R.

Theory

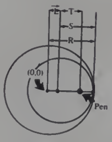
The theory is fairly simple. If you look at Fig. 1 where Sign = 1, you will see a vector L. The length of L is the distance between the center of circles, R and S. L is rotated about the origin, while another Vector T rotates around the end of L. Since the two don't rotate at the same rate, an interesting pattern is created. This can easily be seen in Fig. 2.

In order to save time during execution, I created an array S2 and C2 consisting of sines and cosines. This way, to take a sine or cosine of a number, you only need to execute a memory reference, rather than having the computer do the calculations.

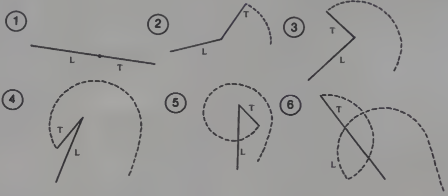
Brian Sietz, 506 Birch Dr., Cherry Hill, NJ 08003.



Sign=1



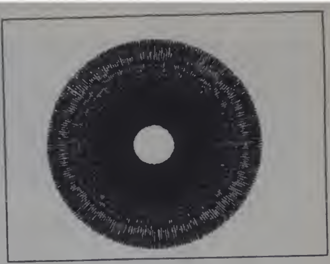
Sign=-1



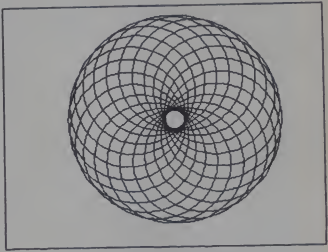
Operation

This program was written on a Hewlett Packard HP9845 desktop computer. It was simplified greatly in order to run on most micros. There are some odd statements that might have to be changed to run on your computer. Line 370 is to enter GRAPHICS mode, line 380 clears the Graphic memory and line 390 draws a border on the screen. Line 440 assigns minimum and maximum for the X-Y coordinates and line 480 determines what portion of the maximum is to be displayed. Most other statements are standard Basic statements with standard variables. This program is made to run on a high resolution screen or a plotter. However, it is possible to modify it to run on lower resolution graphic computers. □

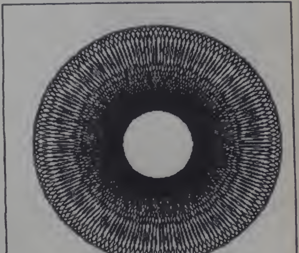
```
10   GOSUB 620
20   DIM S2(80),C2(80)
30   REM INITIALIZE S2 AND C2 ARRAYS
40   REM S2 ARRAY IS AN ARRAY OF SINES
50   REM C2 ARRAY IS AN ARRAY OF COSINES
60   T=80
70   D=2*PI T
80   A=0
90   Q0=T 8
100  Q1=T 4
110  Q2=T 2
120  Q3=3*T 4
130  FOR X=0 TO Q0
140  S2(X)=SIN(A)
150  C2(X)=COS(A)
160  C2(Q1-X)=S2(X
170  S2(Q1  )=C2(X
180  U=X
190  FOR Y=1 TO 2
200  C2(U+Q1)=-S2(U)
210  S2(U+Q2)=-S2 U)
220  C2(U+Q3)=S2(U)
230  S2(U+QI)=C2(U)
240  C2(U+Q2)=-C2(U)
250  S2(U+Q3)=-C2(U)
260  U=QI-X
270  NEXT Y
280  A=A+D
290  NE T X

300  DISP "SELECT R,S,T,SIGN";
310  INPUT R,S,T2,S1
320  REM P IS THE RADIUS OF STATIONARY CIRCLE
330  REM S IS THE RADIUS OF THE MOVING CIRCLE
340  REM T2 IS THE DISTANCE FROM THE CENTER OF CIRCLE S TO THE PEN
350  REM IF S1=1 THEN CIRCLE S ROTATES OUTSIDE CIRCLE R
360  REM IF S1=-1 THEN CIRCLE S ROTATES INSIDE CIRCLE R
370  GPAPHICS
380  GCLEAR
390  FRAME
400  L=R-S1*S
410  REM L IS THE DISTANCE BETWEEN THE CENTERS OF CIRCLES R AND S
420  D1=-S1*R S*D
430  T1=2*PI
440  SCALE -200,200,-200,200
450  X1=(L+T2)*1.1
460  REM X1 IS 10% LARGEP THAN TOTAL RADIUS OF DESIGN
470  REM XI IS USED TO AUTOMATICALLY SCALE THE DESIGN TO THE PAPER
400  SHOW -X1,X1,-X1, 1
490  PENUP
500  X=1
510  P=P1

520  REM PLOTTING ROUTINE
530  PLOT L*C2(  +T2*COS(P ,L*S2(  +T2*SIN(P)
540  P=P+D1
550  IF -S1*P T1 THEN P=P+S1*T1
560  X=X+1
570  IF X<=T THEN 530
580  X=1
590  IF P  -S1*PI THEN 530
600  PLOT L*C2(X)+T2*COS(P),L*S2   +T2*SIN(P
610  END

620  PRINT PAGE
630  PRINT "THIS IS A SPIRO-GRAPH PROGRAM"
640  PRINT "To create a design, we hold a pen a distance T from"
650  PRINT "the center of a circle with radius S, and spin it around "
660  PRINT "a circle with radius R.  By selecting a good ratio for"
670  PRINT "P and S you can get some interesting designs.  The SIGN"
680  PRINT "will be either 1 or -1.  I means than the circle with radius"
690  PRINT "S rotates on the inside of the circle with radius R, and "
700  PRINT "-I means that the circle with radius S spins on the "
710  PRINT "outside of the circle with radius R."
720  RETURN
```



R=153 S=98 T=37 Sign=1



R=5 S=115 T=100 Sign=-1



R=117 S=43 T=41 Sign=1

R=55 S=35 T=30 Sign=1

High-Resolution Graphics For The Sorcerer

Dale M. Gass

For those of you Sorcerer users who need higher resolution than 64x30 for plotting and find when they use higher resolution plotting routines (such as Vic Tolomei's HIRES plot) they run out of graphic characters, here is a compromise. I have written a routine which allows the Sorcerer to plot on a grid 128x60 (similar to the TRS-80) without running out of graphic characters to plot with. Many TRS-80 programs which use plotting (such as in the TRS-80 column TRS-80 Strings can be converted to run on the Sorcerer.

The routine presented in this article is fairly short and resides in the memory locations 0000H-0076H and will fit on any Sorcerer.

Enter the program listed using the standard method described in "A Guided Tour of Personal Computing" which is supplied with every Sorcerer bought.

The plotting routine also requires that five of the programmable graphic characters be defined. The values for the graphic characters are listed below. Enter them in the standard method also.

```
FE00: FF FF FF FF F0 F0 F0 F0
FE08: FF FF FF FF 0F 0F 0F 0F
FE10: F0 F0 F0 F0 FF FF FF FF
FE18: 0F 0F 0F 0F FF FF FF FF
FE20: FF FF FF FF FF FF FF FF
```

Now comes the actual use of the routine. To start poke the values of zero into both locations 260 and 261 with the statement:
10 POKE 260,0: POKE 261,0

Dale Gass, R.R. #1, Brookfield, Nova Scotia, Canada, B0N 1C0.

```
+-------------------------------------------------+
!CONVERSION OF TRS-80 GRAPHICS PROGRAM TO SORCERER!
+-------------------------------------------------+

For original program see 'TRS-80 STRINGS', Creative
Computing, September 1979, page 186.

(NOTE: TRS-80 GRAPHICS SIMULATION PLOTTING ROUTINE MUST
BE IN MEMORY (0000-0076) WHEN USING THIS PROGRAM.) ----

0 GOTO 100
1 POKE 100,1:GOTO 3
2 POKE 100,0
3 POKE 98,ROW:POKE 99,COL
4 Z9=USR(Z9):RETURN
100 CLEAR 100:PRINT CHR$(12);
103 POKE 260,0:POKE 261,0
106 DEF FNR(Z9)=INT(RND(1)*Z9+1)
110 X=FNR(20)
120 Y=FNR(10)
130 COL=X:ROW=Y:GOSUB 1
140 COL=X+1:ROW=Y+1:GOSUB 2
150 COL=41-X:ROW=Y:GOSUB 1
160 COL=40-X:ROW=Y+1:GOSUB 2
170 COL=X:ROW=21-Y:GOSUB 1
180 COL=X+1:ROW=20-Y:GOSUB 2
190 COL=41-X:ROW=21-Y:GOSUB 1
200 COL=40-X:ROW=20-Y:GOSUB 2:
GOTO 110
```

This sets the address of the call routine to the plotting routine at location 0000H.

Next poke the Y co-ordinate value into location 98 and the X co-ordinate value into 99 of the point you wish to plot with the statement:
20 POKE 98, Y:POKE 99,X

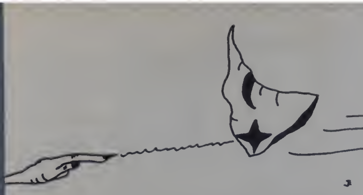
You now have two options: plot a white point at the co-ordinate specified in the last step, or erase a white point at the co-ordinate specified.
To plot a white point use the statement:
30 POKE 100,1

To erase a white point use the statement:
30 POKE 100,0

You can now plot (or erase) the point specified with the statement 'Z9=USRZ9)' or any equivalent call.

Caution: This routine does NOT check for illegal co-ordinates. If you are careless you could cause the routine to poke into vital areas of memory and cause a system crash.

The following is a listing of the plotting routine and a sample program which uses it. □



```
+----------------------------------------+
|SIMULATION OF TRS-80 GRAPHICS ON SORCERER|
+----------------------------------------+

0000: F5            START:PUSH AF
0001: C5                  PUSH BC
0002: D5                  PUSH DE
0003: E5                  PUSH HL
0004: DD E5               PUSH IX
0006: AF                  XOR A
0007: ED 4B 62 00         LD BC,(COORDS)
000B: CB 39               SRL C
000D: 8F                  ADC A,A
000E: CB 38               SRL B
0010: 8F                  ADC A,A
0011: 16 01               LD D,01
0013: B7                  OR A
0014: 28 05               JR Z,P1
0016: CB 22           P2: SLA D
0018: 3D                  DEC A
0019: 20 FB               JR NZ,P2
001B: 67                  LD H,A
001C: 69                  LD L,C
001D: 3E 06               LD A,06
001F: 29              P3: ADD HL,HL
0020: 3D                  DEC A
0021: 20 FC               JR NZ,P3
0023: 48                  LD C,B
0024: 47                  LD B,A
0025: 09                  ADD HL,BC
0026: 01 80 F0            LD BC,F080
0029: 09                  ADD HL,BC
002A: 47                  LD B,A
002B: 4E                  LD C,(HL)
002C: DD 21 65 00         LD IX,TABLE
0030: DD 7E 00        P4: LD A,(IX+00)
0033: B7                  OR A
0034: 28 08               JR Z,S1
0036: B9                  CP C
0037: 28 06               JR Z,S2
0039: 04                  INC B
003A: DD 23               INC IX
003C: 18 F2               JR P4
003E: 47              S1: LD B,A
003F: 3A 64 00            LD A,(MODE)
0042: B7                  OR A
0043: 28 04               JR Z,S3
0045: 78                  LD A,B
0046: B2                  OR D
0047: 18 03               JR S4
0049: 7A              S3: LD A,D
004A: 2F                  CPL
004B: A0                  AND B
004C: 47              S4: LD B,A
004D: DD 21 64 00         LD IX,TABLE-1
0051: 04                  INC B
0052: DD 23           S5: INC IX
0054: 05                  DEC B
0055: 20 FB               JR NZ,S5
0057: DD 7E 00            LD A,(IX+00)
005A: 77                  LD (HL),A
005B: DD E1               POP IX
005D: E1                  POP HL
005E: D1                  POP DE
005F: C1                  POP BC
0060: F1                  POP AF
0061: C9                  RET
```

BY Dale Gass (1980)

THIS IS A TABLE REFERRED TO BY THE PROGRAM AND MUST BE IN THE PROPER MEMORY LOCATION WHEN RUNNING.

```
0065: 20 A1 A0 A9 96 A7 A6 C0 95 A5 A8 C1 AA C2 C3 C4 00
```

Low-Resolution Graphics for the Sorcerer


Photo 1. Sorcerer hi-resolution graphics.

Bob Stuckmeyer

After watching the evolution of personal computers over the past six years, I recently decided to "grab for some of the gusto" by purchasing an Exidy Sorcerer. At $1295 for the standard machine, the Sorcerer is a little more expensive than comparable systems from Apple, Commodore or Radio Shack. I was willing to spend the extra money because I felt the Sorcerer offered greater value per investment dollar.

The Sorcerer has a Z-80 cpu and the standard unit comes with 16K of memory. Memory is internally expandable to 48K. The keyboard has a professional feel and consists of 63 keys plus a 16-key numeric pad. Other standard features include an RS-232 serial interface which can be used to control two cassette recorders, a parallel interface compatible with Centronics printers, a video interface, and an expansion bus. The 1200 baud cassette interface is one of the most reliable I have seen; in five months of use I have had only two programs fail to load on the first attempt.

Two features that I believe give the Sorcerer an advantage over the competition are its use of plug-in ROM software packs and its graphics capabilities. The computer comes with 8K Microsoft Basic in ROM packaged in a device resembling an 8-track stereo tape. This device is plugged into the right side of the keyboard unit. If programming in Basic is not your cup of tea, you can purchase a cartridge containing assembler or one containing a word processor.

Having acquainted you with some of the Sorcerer's features, I can now proceed to the main topic of this article—the graphics capabilities of the computer. The video screen consists of 30 lines of 64 characters. Each character is represented as an 8X8 matrix of dots, yielding an effective resolution of 512 (64X8) horizontal by 240 (30X8) vertical points. The representation of each character is contained in 8 successive memory locations. Thus 2,048 memory locations are used to represent all 256 ASCII codes. The user can define up to 128 of his own characters by altering the contents of the memory locations assigned to ASCII codes 128-255.

Bob Stuckmeyer, 2347 Cavendish Lane, St. Louis, MO 63129.

I'll illustrate how to define characters by means of an example. Let's say that you wanted to display an upside down letter "A" on the screen. Figure 1 contains the dot matrix representation of this character.

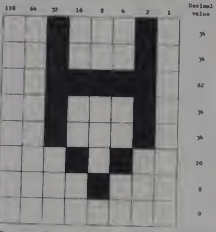


Figure 1. Dot matrix representation of upside down "A".

The following Basic statements assign this character to ASCII code 192 (arbitrary choice). Once the above statements are executed, the character could be displayed on the screen by the command PRINT CHR$(192) or by POKEing 192 into the desired video RAM location.

```
10 REM MEMORY ADDRESSES -512 TO -505 CONTAIN THE
20 REM REPRESENTATION OF ASCII CODE 192
30 FOR I=-512 TO -505
40 READ X:POKE I,X
50 NEXT I
60 DATA 34,34,62,34,34,20,8,0
```

High-resolution graphics are accomplished by using this technique to define multiple graphic characters. A character might only represent a piece of a figure, and hence must be combined with other characters on the screen to display a complete figure. The figure in Photo 1 was constructed in this manner and required the definition of 71 unique characters.

The process of defining and combining characters to create graphics is tedious, and ill-suited for applications like arithmetic function plotting or displaying the path of a photon torpedo. For these applications, a much simpler approach would be to specify a point's x and y coordinates along with a command to turn on the point. Similar commands would be required to turn off a point and to test a point to see if it is on or off. Level II Basic on Radio Shack's TRS-80 provides SET, RESET, and POINT commands which accomplish these functions.

The subroutines contained in Listing 1 will simulate these commands at the expense of resolution. Under the approach adopted, the ability to manipulate any of the 521X240 points would have required definition of enough graphic characters to account for all combinations of points on or off within a character. Since each character contains 64 dots, there are two possible representations (clearly exceeding the Sorcerer's capabilities). If each character is instead viewed as containing four points which can be on or off, only two graphics characters are necessary to account for all combinations of points (see figure 2). This approach yields a resolution of 128 horizontal by 60 vertical points slightly greater than that provided on the TRS-80.

The story behind the two best selling computer games books in the world.

Computer Games

by David H. Ahl

Everybody likes games. Children like tic tac toe. Gamblers like blackjack. Trekkies like Star Trek. Almost everyone has a favorite game or two.

It Started in 1971

Ten years ago when I was at Digital Equipment Corp. (DEC), we wanted a painless way to show reluctant educators that computers weren't scary or difficult to use. Games and simulations seemed like a good method.

So I put out a call to all our customers to send us their best computer games. The response was overwhelming. I got 21 versions of blackjack, 15 of nim and 12 of battleship.

From this enormous outpouring I selected the 90 best games and added 11 that I had written myself for a total of 101. I edited these into a book called 101 Basic Computer Games which was published by DEC. It still is.

When I left DEC in 1974 I asked for the rights to print the book independently. They agreed as long as the name was changed.

Converted to Microsoft Basic

The games in the original book were in many different dialects of Basic. So Steve North and I converted all the games to standard Microsoft Basic, expanded the descriptions and published the book under the new name Basic Computer Games.

Over the next three years, people sent in improved versions of many of the games along with scores of new ones. So in 1979, we totally revised and corrected Basic Computer Games and published a completely new companion volume of 84 additional games called More Basic Computer Games. This edition is available in both Microsoft Basic and TRS-80 Basic for owners of the TRS-80 computer.

Today Basic Computer Games is in its fifth printing and More Basic Computer Games is in its second. Combined sales are over one half million copies making them the best selling pair of books in recreational computing by a wide margin. There are many imitators, but all offer a fraction of the number of games and cost far more.

The games in these books include classic board games like checkers. They include challenging simulation games like Camel (get across the desert on your camel) and Super Star Trek. There are number games like Guess My Number, Stars and Battle of Numbers. You'll find gambling games like blackjack, keno, and poker. All told there are 185 different games in these two books.

Whether you're just getting started with computers or a proficient programmer, you'll find something of interest. You'll find 15-line games and 400-line games and everything in between.

The value offered by these books is outstanding. Every other publisher has raised the price of their books yet these sell for the same price as they did in 1974.

Moneyback Guarantee

Examine one or both of these books and key some games into your computer. If you're not completely satisfied we'll refund the full purchase price plus your return postage.

Basic Computer Games costs only \$7.50 and More Basic Computer Games just \$7.95 for either the Microsoft or TRS-80 edition (please specify your choice on your order). Both books together are \$15. Send payment plus \$2.00 shipping and handling to Creative Computing Press, Morris Plains, NJ 07950. Visa, MasterCard and American Express orders should include card number and expiration date. Charge card orders may also be called in toll-free to 800-631-8112 (In NJ 201-540-0445).

Order today to turn your computer into the best game player on the block.

Contents of *Basic Computer Games* (right) and *More Basic Computer Games* (below).

Artillery-3
Baccarat
Bible Quiz
Big 6
Binary
Blackbox
Bobstones
Bocce
Boga II
Bumbrun
Bridge-It
Camel
Chase
Chuck-A-Luck
Close Encounters
Column
Concentration
Condot
Convoy
Corral
Countdown
Cup
Dealer's Choice
Deepspace
Defuse
Dodgem
Doors
Drag
Dr. Z
Eliza
Father
Flip
Four In A Row
Geowar
Grand Prix
Guess-It
ICBM
Inkblot
Joust
Jumping Balls
Keno
L Game

Life Expectancy
Lissajous
Magic Square
Man-Eating Rabbit
Maneuvers
Mastermind
Masterbagels
Matpuzzle
Maze,
Millionaire
Minotaur
Motorcycle Jump
Nomad
Not One
Obstacle
Octrix
Pasart
Pasart 2
Pinball
Rabbit Chase
Roadrace
Rotate
Safe
Scales
Schmoo
Seabattle
Seawar
Shoot
Smash
Strike 9
Tennis
Tickertape
TV Plot
Twonky
Two-to-Ten
UFO
Under & Over
Van Gam
Warfish
Word Search Puzzle
Wumpus 1
Wumpus 2

Introduction
The Basic Language
Conversion to Other Basics
Acey Ducey
Amazing
Animal
Awari
Bagels
Banner
Basketball
Batnum
Battle
Blackjack
Bombardment
Bombs Away
Bounce
Bowling
Boxing
Bug
Bullfight
Bullseye
Bunny
Buzzword
Calendar
Change
Checkers
Chemist
Chief
Chomp
Civil War
Combat
Craps
Cube
Depth Charge
Diamond
Dice
Digits
Even Wins
Flip Flop
Football
Fur Trader
Golf
Gomoko
Guess
Gunner
Hammurabi
Hangman
Hello
Hexapawn

Hi-Lo
High I-Q
Hockey
Horserace
Hurkle
Kinema
King
Letter
Life
Life For Two
Literature Quiz
Love
Lunar LEM Rocket
Master Mind
Math Dice
Mugwump
Name
Nicomachus
Nim
Number
One Check
Orbit
Pizza
Poetry
Poker
Queen
Reverse
Rock, Scissors, Paper
Roulette
Russian Roulette
Salvo
Sine Wave
Slalom
Slots
Splat
Stars
Stock Market
Super Star Trek
Synonym
Target
3-D Plot
3-D Tic-Tac-Toe
Tic Tac toe
Tower
Train
Trap
23 Matches
War
Weekday
Word

creative computing

Morris Plains, NJ 07950
Toll-free ***800-631-8112***
(In NJ 201-540-0445)

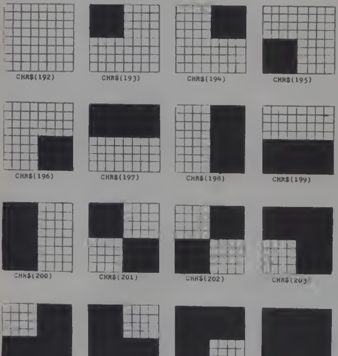


Figure 2. Graphics characters used by Listing 1 to represent all combinations of four points on or off.

This approach yields a resolution of 128 horizontal by 60 vertical points.

Listing 1

```
10 REM ** SORCERER LOW-RESOLUTION GRAPHICS DEMO **
15 REM **                BY                       **
20 REM **          BOB STUCKMEYER                 **
25 REM **          2347 CAVENDISH LN              **
30 REM **          ST. LOUIS, MO 63129            **
40 REM
50 REM ** THIS PROGRAM ILLUSTRATES USE OF LOW     **
55 REM ** RESOLUTION GRAPHICS SUBROUTINES BY      **
60 REM ** MOVING A POINT IN RANDOM DIRECTIONS.    **
65 REM ** AS THE POINT IS MOVED TO EACH NEW LO-   **
70 REM ** CATION THE LOCATION IS TURNED ON IF     **
75 REM ** IT IS CURRENTLY OFF AND TURNED OFF      **
80 REM ** OTHERWISE.                              **
85 REM
100 PRINT CHR$(12);CHR$(23);:GOSUB 7000
110 R=RND(-1):X=INT(RND(1)*128)
120 Y=INT(RND(1)*60):GOSUB 8000
130 R=INT(RND(1)*8)+1
140 ON R GOSUB 300,320,340,360,380,400,420,440
150 IF X>127 THEN X=0
160 IF X<0 THEN X=127
170 IF Y>59 THEN Y=0
180 IF Y<0 THEN Y=59
190 GOSUB 8800
200 IF P=1 THEN GOSUB 8400:GOTO 220
210 GOSUB 8000
220 GOTO 130
300 Y=Y+1:RETURN
320 X=X+1:Y=Y+1:RETURN
340 X=X+1:RETURN
360 X=X+1:Y=Y-1:RETURN
380 Y=Y-1:RETURN
400 X=X-1:Y=Y-1:RETURN
420 X=X-1:RETURN
440 X=X-1:Y=Y+1:RETURN
7000 REM ** SUBROUTINE THAT DEFINES GRAPHICS CHARACTERS **
7010 REM ** USED TO SET AND RESET POINTS ON THE SCREEN  **
7020 FOR I=-512 TO -385:READ Z:POKE I,Z:NEXT I
7030 DATA 0,0,0,0,0,0,0,0                        :REM CHR$(192)
7040 DATA 240,240,240,240,0,0,0,0                :REM CHR$(193)
7050 DATA 15,15,15,15,0,0,0,0                    :REM CHR$(194)
7060 DATA 0,0,0,0,240,240,240,240                :REM CHR$(195)
7070 DATA 0,0,0,0,15,15,15,15                    :REM CHR$(196)
7080 DATA 255,255,255,255,0,0,0,0                :REM CHR$(197)
7090 DATA 15,15,15,15,15,15,15,15                :REM CHR$(198)
7100 DATA 0,0,0,0,255,255,255,255                :REM CHR$(199)
7110 DATA 240,240,240,240,240,240,240,240        :REM CHR$(200)
7120 DATA 240,240,240,240,15,15,15,15            :REM CHR$(201)
7130 DATA 15,15,15,15,240,240,240,240            :REM CHR$(202)
7140 DATA 255,255,255,255,15,15,15,15            :REM CHR$(203)
7150 DATA 15,15,15,15,255,255,255,255            :REM CHR$(204)
7160 DATA 240,240,240,240,255,255,255,255        :REM CHR$(205)
7170 DATA 255,255,255,255,240,240,240,240        :REM CHR$(206)
7180 DATA 255,255,255,255,255,255,255,255        :REM CHR$(207)
7190 RETURN
8000 REM ** SUBROUTINE THAT SETS (TURNS ON) THE GRAPHICS **
8010 REM ** BLOCK GIVEN BY THE POINT (X,Y).  X IS THE    **
8020 REM ** BLOCK'S HORIZONTAL POSITION (0 THRU 127).  Y **
8030 REM ** IS THE BLOCK'S VERTICAL POSITION (0 THRU 59).**
8040 REM ** THE POINT (0,0) IS THE LOWER LEFT HAND COR-  **
8050 REM ** NER OF THE SCREEN.                           **
8060 REM
8070 REM ** 1)  RANGE CHECK **
8080 IF (X<0 OR X>127 OR Y<0 OR Y>59) THEN 8280
8090 X=INT(X):Y=INT(Y)
8100 REM **  2)  DETERMINE MEMORY ADDRESS OF CHARACTER **
8110 REM **      CONTAINING THE POINT (X,Y), AND MATCH **
8120 REM **      CHARACTER AT ADDRESS WITH PREVIOUSLY  **
8130 REM **      DEFINED GRAPHICS CHARACTERS           **
8150 GOSUB 9000
8160 REM **  3)  TURN ON GRAPHICS BLOCK (IF IT IS OFF) **
8180 J=-384
8190 IF INT(Y/2)=Y/2 THEN J=-380
8200 IF INT(X/2)=X/2 THEN 8240
8210 IF (PEEK(J)=15 OR PEEK(J)=255) THEN 8280
```

```
8220 FDR I=J TO J+3:PDKE I,PEEK(I)+15:NEXT I
8230 GDTO 8260
8240 IF PEEK(J)>15 THEN 8280
8250 FDR I=J TO J+3:PDKE I,PEEK(I)+240:NEXT I
8260 REM ** 4)  STDRE UPDATED CHARACTER **
8270 GDSUB 9400
8280 RETURN
8400 REM ** SUBROUTINE THAT RESETS (TURNS DFF) THE GRAPHICS **
8410 REM ** BLOCK GIVEN BY THE POINT (X,Y).                 **
8420 REM
8430 REM ** 1)  RANCE CHECK **
8440 IF (X<0 OR X>127 DR Y<0 DR Y>59) THEN 8670
8450 X=INT(X):Y=INT(Y)
8460 REM **  2)  DETERMINE MEMDRY ADDRESS DF CHARACTER **
8470 REM **      CONTAINING THE PDINT (X,Y), AND MATCH **
8480 REM **      CHARACTER AT ADDRESS WITH PREVIDUSLY  **
8490 REM **      DEFINED GRAPHICS CHARACTERS           **
8510 GDSUB 9000
8560 REM **  3)  TURN OFF GRAPHICS BLDCK (IF IT IS DN) **
8570 J=-384
8580 IF INT(Y/2)=Y/2 THEN J=-380
8590 IF INT(X/2)=X/2 THEN 8630
8600 IF (PEEK(J)=0 OR PEEK(J)=240) THEN 8670
8610 FOR I=J TO J+31POKE I,PEEK(I)-15:NEXT I
8620 GOTD 8650
8630 IF PEEK(J)<16 THEN 8670
8640 FOR I=J TO J+3:POKE I,PEEK(I)-240:NEXT I
8650 REM **  4)  STDRE UPDATED CHARACTER **
8660 GDSUB 9400
8670 RETURN
8800 REM ** SUBRDUTINE THAT DETERMINES IF THE GRAPHICS   **
8810 REM ** BLDCK GIVEN BY THE PDINT (X,Y) IS DN DR OFF. **
8820 REM ** THE VARIABLE P IS ASSIGNED A VALUE OF 0 TO   **
8830 REM ** INDICATE 'DFF' AND A VALUE OF 1 FOR 'ON'.    **
8840 P=0
8850 REM **  1)  RANGE CHECK **
8860 IF (X<0 DR X>127 OR Y<0 DR Y>59) THEN 8995
8870 X=INT(X):Y=INT(Y)
8880 REM **  2)  DETERMINE MEMORY ADDRESS DF CHARACTER **
8890 REM **      CONTAINING THE PDINT (X,Y), AND MATCH **
8900 REM **      CHARACTER AT ADDRESS WITH PREVIDUSLY  **
8910 REM **      DEFINED GRAPHICS CHARACTERS           **
8930 GDSUB 9000
8940 REM **  3)  SEE IF GRAPHICS BLDCK IS DN **
8950 J=-384
8960 IF INT(Y/2)=Y/2 THEN J=-380
8970 IF INT(X/2)=X/2 THEN 8990
8980 IF (PEEK(J)=15 OR PEEK(J)=255) THEN P=1:COTO 8995
8985 GDTD 8995
8990 IF PEEK(J)>15 THEN P=1
8995 RETURN
9000 REM ** SUBRDUTINE THAT CDMPUTES MEMDRY ADDRESS DF **
9010 REM ** THE CHARACTER CDNTAINING THE PDINT (X,Y),  **
9020 REM ** MATCHES THE CHARACTER WITH THE PREVIDUSLY  **
9030 REM ** DEFINED GRAPHICS CHARACTERS.  DNCE THE     **
9035 REM ** CHARACTER IS MATCHED IT'S GRAPHIC REPRE-   **
9040 REM ** SENTATION IS CDPIED TD THE MEMORY LDCA-    **
9050 REM ** TIDNS WHICH DEFINE THE REPRESENTATION OF   **
9060 REM ** CHR$(208).  IF NO MATCH IS FDUND THE CHAR- **
9070 REM ** ACTER IS ASSUMED TO MATCH CHR$(192) (ALL   **
9073 REM ** PDINTS DFF).                               **
9077 AD=INT(X/2)-64*(INT(Y/2))-2112
9080 FDR I=-384 TD -377:PDKE I,0:NEXT I
9090 FDR I=192 TO 207
9100 IF CHR$(PEEK(AD))<>CHR$(I) THEN 9140
9110 J=-512+((I-192)*8)
9120 FDR K=-384 TD -377:PDKE K,PEEK(J):J=J+1:NEXT K
9130 GOTO 9150
9140 NEXT I
9150 RETURN
9400 REM ** SUBRDUTINE WHICH STDRES THE UPDATED CHARACTER **
9410 REM **                                               **
9420 REM ** UPON ENTRY THE UPDATED CHARACTER'S GRAPHIC    **
9430 REM ** REPRESENTATION IS CONTAINED IN THE MEMORY     **
9440 REM ** LOCATIONS WHICH DEFINE THE REPRESENTATION OF  **
9450 REM ** CHR$(208).  THIS REPRESENTATION IS COMPARED   **
9460 REM ** WITH THE REPRESENTATIONS OF CHR$(192) THRU    **
9470 REM ** CHR$(207).  WHEN A MATCH IS FOUND THE UPDATED **
9480 REM ** CHARACTER IS STORED BY POKING THE ASCII VALUE **
9490 REM ** OF THE MATCHING CHARACTER INTO THE MEMORY AD- **
9500 REM ** DRESS DF THE PDINT (X,Y).                     **
9510 FDR I=-512 TO -392 STEP 8
9520 FDR J=0 TD 7
9530 IF PEEK(I+J)<>PEEK(-384+J) THEN 9570
9540 NEXT J
9550 POKE AD,(I+512)/8+192
9560 GOTO 9580
9570 NEXT I
9500 RETURN
```


Although the program in Listing 1 is fairly well documented, several points may require further clarification:

1) The colon (:) serves as a command delimiter.

2) PRINT CHR$(12); CHR$(23); will clear the screen and uppress the cursor.

3) RND (N) will start a different sequence of random numbers if N is negative, and will produce numbers between 0 and 1 if N is a positive interger.

4) Character representations of ASCII codes 192 thru - 208 resideat addresses -512 thru -377.

5) Because the author's academic background was in mathematics, the four corners of the screen were assigned the coordinates depicted in Figure 3(pay attention, Radio Shack!).

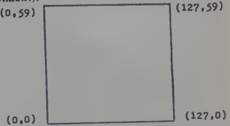


Figure 3. Screen coordinates.

Since these routines were written in Basic they run rather slowly. An obvious enhancement would be a machine-language implementation, accessible from Basic. □

The Sorcerer Meets the Paper Tiger

Bob Stuckmeyer

8.3 characters per inch.
8.3 characters per inch, enhanced.
10 characters per inch.
10 characters per inch, enhanced.
12 characters per inch.
12 characters per inch, enhanced.
16.5 characters per inch.
16.5 characters per inch, enhanced.

Figure 1. Paper Tiger character sizes.

While I was able to select a personal computer only after several months of deliberation, choosing a printer to use with it was on the other hand, quite straightforward. The main features I required in a printer were: upper/lower case; an interface compatible with my Sorcerer's parallel port; and most importantly, the ability to reproduce any graphics displayed on the Sorcerer's video screen. Only one printer, the "Paper Tiger" from Integral Data Sytstems, met all of these requirements while still remaining in an affordable price range.

With a list price of $995 and a variety of features, the Paper Tiger is one of the best values in the dot matrix printer market. Standard features include serial RS-232-C and parallel Centronics-compatible interfaces, adjustable pin-feed tractors which support paper widths from 1.75 to 9.5 inches, four character density (see Figure 1), and built-in diagnostics. For an additional $99 a 2K buffer and graphics plotting mode can be added. The graphics option allows the printer to be used for printing illustrations, graphics, and charts; hence it met my main requirement of printing Sorcerer graphics.

The Paper Tiger enters graphics mode upon receipt of an ETX character (hex 03). In this mode the printer uses a raster-scan technique to print columns of vertical dots during each pass of the print head. Up to seven dots can be printed in any column (see Figure 2). Because bit 6 of tions of a received data character determine which positions are printed in each column (see Figure 2). Because bit 6 of each pass is overwritten by bit 0 of the next pass, bit 6 of each character should be set to zero—resulting in only 6 dots

Figure 2. Printer Operation in Graphics Mode.

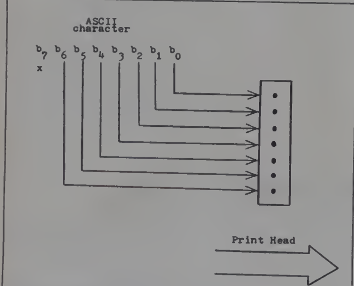
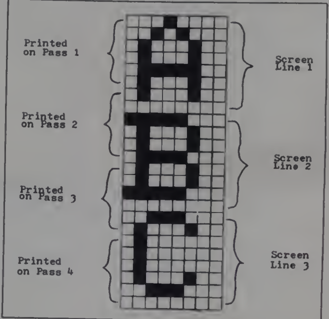


Figure 3. Printing Sorcerer Characters in Graphics Mode.



Bob Stuckmeyer, 2347 Cavendish Lane, St. Louis, MO 63129.

being printed on a pass.

Prior to discussing software for interfacing the Sorcerer with the Paper Tiger, a few of the Sorcerer's characteristics need explanation. The video screen consists of thirty lines of sixty-four characters. Characters are displayed as an 8X8 matrix of dots. Each ASCII character's representation is defined in eight bytes of memory. Graphic characters can be created by altering the representations of ASCII codes 128-255. Pictures are generated by defining and combining multiple graphic characters.

As one might expect, a problem arises when trying to print 8x8 graphic characters on a printer that prints six vertical dots per pass. Figure 3 illustrates this situation, in which 4 passes of the print head are required to print 3 screen lines. Another interface consideration is that the Paper Tiger prints a maximum of 498 horizontal dots per line—thus only allowing 62 of the Sorcerer's 64 characters to be printed per line.

Listing 1 is a Z-80 assembly language program that allows the Paper Tiger to function as the Sorcerer's "screen printer". The program is a subroutine which, when invoked, will duplicate on the printer any text or graphics that appears on the video. Figure 4 was created by calling this subroutine from Creative Computing's Lunar Lander game.

Finally, for those Sorcerer owners who may attempt to use this program, several points should be kept in mind:

1) The folks at Exidy did not equip their computer with a sure-fire method of protecting machine language subroutines from bing clobbered by the Basic interpreter (the Sorcerer has no feature comparable to the TRS-80's MEMORY SIZE?)

2) The Sorcerer software manual suggests putting machine language subroutines in Basic freespace. Unfortunately freespace is dynamic; its location varies depending on the total amount of RAM, the program size and the amount of string space cleared by the program.

3) The program in Listing 1 resides in Basic freespace on a 16K machine. Consequently, it will probably have to be relocated to run on a Sorcerer with a different size memory.

4) I am willing to provide anyone having a Sorcerer Development Pac a cassette containing the source assembler language program (for a nominal charge). □

References

1. *Paper Tiger IDS-440 Impact Printer Owner's Manual*, Integral Data Systems Inc., 14 Tech Circle, Natick, MA 01760.

2. *Sorcerer Software Manual*, Exidy Incorporated, 390 Java Drive, Sunnyvale, CA 94086.

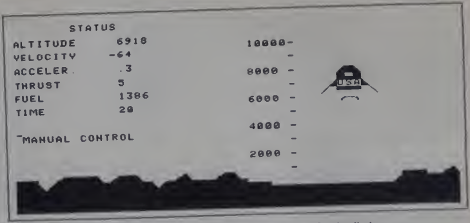


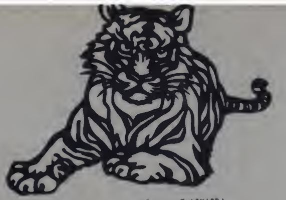
Figure 4. Print of Creative Computing's Lunar Lander display.

Listing 1

```
EXIDY Z-80 ASSEMBLER
  ADDR  OBJECT   ST #

                 0001 ;       EXIDY SORCERER SCREEN PRINTER INTERFACE
                 0002 ;       FOR THE IDS-440 PAPER TIGER
                 0003 ;             BY BOB STUCKMEYER
                 0004 ;             2347 CAVENDISH LANE
                 0005 ;             ST LOUIS, MO 63129
                 0006 ;                 03/25/80
                 0007 ;*********************************************
                 0008 ;
                 0009 ;       ASCII EQUATES
                 0010 ;
 >0003           0011 ETX     EQU     03H     ;GRAPHICS MODE CONTROL CHAR
 >000C           0012 FF      EQU     0CH     ;FORM FEED
 >000B           0013 VT      EQU     0BH     ;VERTICAL TAB
 >0011           0014 DC1     EQU     11H     ;SELECT PRINTER
 >0013           0015 DC3     EQU     13H     ;DESELECT PRINTER
 >0002           0016 STX     EQU     02H     ;NORMAL MODE CONTROL CHAR
 >001E           0017 RS      EQU     1EH     ;SELECT 12 CHAR PER INCH
                 0018 ;
                 0019 ;       RAM POINTERS
                 0020 ;
 >F080           0021 SCREEN  EQU     0F080H
 >F800           0022 CHRDEF  EQU     0F800H  ;ASCII CHAR REPRESENTATIONS
                 0023 ;
                 0024         PSECT   ABS
                 0025         ORG     3856H
                 0026 ;
                 0027 ;       GENERAL INITIALIZATION
                 0028 ;
 3856  F5        0029 SPRINT  PUSH    AF      ;WE DESTROY THESE
 3857  C5        0030         PUSH    BC
 3858  E5        0031         PUSH    HL
 3859  D5        0032         PUSH    DE
 385A  FDE5      0033         PUSH    IY
 385C  DDE5      0034         PUSH    IX
 385E  2180F0    0035         LD      HL,SCREEN
                 0036 ;
                 0037 ;       PRINTER INITIALIZATION
                 0038 ;
 3861  3E11      0039         LD      A,DC1   ;SELECT PRINTER
 3863  CD7D3A    0040         CALL    CENOUT
 3866  3E1E      0041         LD      A,RS    ;SELECT 12 CPI
 3868  CD7D3A    0042         CALL    CENOUT
 386B  3E03      0043         LD      A,ETX   ;ENTER GRAPHICS MODE
 386D  CD7D3A    0044         CALL    CENOUT
                 0045 ;
                 0046 ;       MAIN PROGRAM
                 0047 ;
 3870  060A      0048         LD      B,10D
 3872  CD9938    0049 MLOOP   CALL    PLINE   ;PRINT NEXT 3 LINES
 3875  1600      0050         LD      D,0D
 3877  1EC0      0051         LD      E,192D
 3879  19        0052         ADD     HL,DE   ;POINT TO NEXT 3 LINES
 387A  10F6      0053         DJNZ    MLOOP-$
                 0054 ;
                 0055 ;       PROGRAM WINDUP
                 0056 ;
 387C  3E03      0057         LD      A,ETX
 387E  CD7D3A    0058         CALL    CENOUT
 3881  3E02      0059         LD      A,STX   ;RETURN TO NORMAL MODE
 3883  CD7D3A    0060         CALL    CENOUT
 3886  3E0C      0061         LD      A,FF    ;ISSUE FORM FEED
```

Paper Tiger, continued...

```
3888  CD7D3A          CALL   CENOUT
388B  3EI3            LD     A,DC3    ;DESELECT PRINTER
388D  C07D3A          CALL   CENOUT
3890  00EI            POP    IX
3892  FDE1            POP    IY
3894  D1              POP    DE
3895  EI              POP    HL
3896  CI              POP    BC
3897  FI              POP    AF
3898  C9              RET
                ;     SUBROUTINE THAT PRINTS 3 SCREEN LINES.
                ;     BECAUSE THE SORCERER'S CHARACTERS ARE BXB
                ;     ANO THE PAPER TIGER ONLY PRINTS 6 VERTICAL
                ;     OOTS PER PASS, 4 PASSES ARE REQUIRED TO
                ;     PRINT 3 LINES.  PASS I PRINTS THE TOP 6 DOTS
                ;     OF THE FIRST LINE.  PASS 2 PRINTS THE LAST 2
                ;     DOTS OF THE FIRST LINE AND THE TOP 4 DOTS OF
                ;     THE SECOND LINE AND SO ON.
                ;
3899  C5        PLINE   PUSH   BC       ;WE DESTROY
389A  E5                PUSH   HL
389B  ODEI              POP    IX       ;POINT IX TO CURRENT LINE
3890  063E              LD     B,62D    ;LOOP FOR 62 CHARS PER LINE
389F  CDDA38    PLOOPI  CALL   T6       ;PRINT TOP 6 DOTS
38A2  DD23              INC    IX
38A4  IOF9              DJNZ   PLOOP1-$
38A6  CD383A            CALL   NEWLN    ;START NEW LINE
38A9  E5                PUSH   HL
38AA  ODEI              POP    IX       ;POINT IX BACK
38AC  063E              LO     B,620
38AE  CDOE39    PLOOP2  CALL   B2T4
38B1  0D23              INC    IX
38B3  IOF9              DJNZ   PLOOP2-$
38B5  CD3B3A            CALL   NEWLN    ;START NEW LINE
3BBB  0D23              INC    IX       ;SKIP LAST TWO CHARS ON SCREEN
38BA  0023              INC    IX
38BC  DDE5              PUSH   IX       ;SAVE ADOR OF CURRENT LINE
3BBE  063E              LO     B,62D
38C0  CD6F39    PLOOP3  CALL   B4T2
38C3  DD23              INC    IX
38C5  IOF9              DJNZ   PLOOP3-$
38C7  CD383A            CALL   NEWLN
38CA  DDEI              POP    IX
38CC  063E              LD     B,62D
38CE  CD0039    PLOOP4  CALL   B6
3B0I  0023              INC    IX
38D3  IOF9              DJNZ   PLOOP4-$
3BD5  CD3B3A            CALL   NEWLN
38D8  C1                POP    BC       ;RESTORE
38D9  C9                RET
                ;
                ;     SUBROUTINE WHICH PRINTS TOP 6 DOTS OF
                ;     A CHARACTER
                ;
38DA  E5        T6      PUSH   HL       ;WE DESTROY
38DB  C5                PUSH   BC
38DC  CDI53A            CALL   INITCHR ;ZERO WORK CHARS
38DF  2I00FB            LD     HL,CHROEF
38E2  1600              LD     D,00
38E4  DD5E00            LD     E,(IX+0D) ;LOAO CHAR IN E
3BE7  CD0B3A            CALL   MPYB     ;MULTIPLY BY 8
3BEA  19                ADO    HL,DE    ;POINT TO CHAR'S REPRESENTATION
38EB  1E00              LO     E,OD     ;LOOP FOR 6 VERT OOTS
38E0  060B      T6LPI   LD     B,80     ;LOOP FOR B HORZ DOTS
38EF  7E                LD     A,(HL)
38F0  F02I943A          LO     IY,WKCHR
3BF4  CB7F      T6LP2   BIT    7,A
38F6  2803              JR     Z,T6CNT-$
38F8  C0433A            CALL   SETBIT
38FB  CB07      T6CNT   RLC    A
38F0  FD23              INC    IY
38FF  IOF3              DJNZ   T6LP2-$
390I  23                INC    HL
3902  IC                INC    E
3903  7B                LD     A,E
3904  FE06              CP     60
3906  20E5              JR     NZ,T6LPI-$
3908  C0203A            CALL   CHROUT   ;OUTPUT WORK CHARS
390B  C1                POP    BC
390C  EI                POP    HL
3900  C9                RET
                ;
                ;     SUBROUTINE WHICH PRINTB BOTTOM 2 DOTS OF A
                ;     CHAR ANO TOP 4 DOTS OF CHAR ON NEXT LINE.
                ;
390E  E5        B2T4    PUSH   HL       ;WE OESTROY
390F  C5                PUSH   BC
39I0  C0I53A            CALL   INITCHR
39I3  2I00FB            LD     HL,CHROEF
3916  I600              LD     D,OD
```



```
3918  DD5E00            LD     E,(IX+0D)
391B  CD083A            CALL   MPY8
391E  7B                LD     A,E
391F  C606              ADD    A,6D
3921  5F                LD     E,A
3922  19                ADD    HL,DE    ;POINT TO CHAR'S REPRESENTATION
3923  1E00              LD     E,0D     ;LOOP FOR 2 VERT DOTS
3925  0608     B2LP1    LD     B,8D     ;LOOP FOR 8 HORZ DOTS
3927  7E                LD     A,(HL)
3928  FD21943A          LD     IY,WKCHR
392C  CB7F     B2LP2    BIT    7,A
392E  2803              JR     Z,B2CNT-$
3930  CD433A            CALL   SETBIT
3933  CB07     B2CNT    RLC    A
3935  FD23              INC    IY
3937  10F3              DJNZ   B2LP2-$
3939  23                INC    HL
393A  1C                INC    E
393B  7B                LD     A,E
393C  FE02              CP     2D
393E  20E5              JR     NZ,B2LP1-$
3940  2100F8            LD     HL,CHRDEF
3943  1600              LD     D,0D
3945  DD5E40            LD     E,(IX+64D)       ;LOAD CHAR ON NEXT LINE
3948  CD083A            CALL   MPY8
394B  19                ADD    HL,DE
394C  1E02              LD     E,2D     ;LOOP FOR 4 VERT DOTS
394E  0608     T4LP1    LD     B,8D     ;LOOP FOR 8 HORZ DOTS
3950  7E                LD     A,(HL)
3951  FD21943A          LD     IY,WKCHR
3955  CB7F     T4LP2    BIT    7,A
3957  2803              JR     Z,T4CNT-$
3959  CD433A            CALL   SETBIT
395C  CB07     T4CNT    RLC    A
395E  FD23              INC    IY
3960  10F3              DJNZ   T4LP2-$
3962  23                INC    HL
3963  1C                INC    E
3964  7B                LD     A,E
3965  FE06              CP     6D
3967  20E5              JR     NZ,T4LP1-$
3969  CD203A            CALL   CHROUT  ;OUTPUT WORK CHARS
396C  C1                POP    BC      ;RESTORE
396D  E1                POP    HL
396E  C9                RET
                       ;
                       ;        SUBROUTINE WHICH PRINTS BOTTOM 4 DOTS OF
                       ;        A CHAR AND TOP 2 DOTS OF CHAR ON NEXT LINE
                       ;
396F  E5       B4T2     PUSH   HL       ;WE DESTROY
3970  C5                PUSH   BC
3971  CD153A            CALL   INITCHR
3974  2100F8            LD     HL,CHRDEF
3977  1600              LD     D,0D
3979  DD5E00            LD     E,(IX+0D)
397C  CD083A            CALL   MPY8
397F  7B                LD     A,E
3980  C604              ADD    A,4D
3982  5F                LD     E,A
3983  19                ADD    HL,DE    ;POINT TO CHAR'S REPRESENTATION
3984  1E00              LD     E,0D     ;LOOP FOR 4 VERT DOTS
3986  0608     B4LP1    LD     B,8D     ;LOOP FOR 8 HORZ DOTS
3988  7E                LD     A,(HL)
3989  FD21943A          LD     IY,WKCHR
398D  CB7F     B4LP2    BIT    7,A
398F  2803              JR     Z,B4CNT-$
3991  CD433A            CALL   SETBIT
3994  CB07     B4CNT    RLC    A
3996  FD23              INC    IY
3998  10F3              DJNZ   B4LP2-$
399A  23                INC    HL
399B  1C                INC    E
399C  7B                LD     A,E
399D  FE04              CP     4D
399F  20E5              JR     NZ,B4LP1-$
```

Paper Tiger, continued...

```
39A1  2100F8            LD     HL,CHRDEF
39A4  1600              LD     D,0D
39A6  DD5E40            LD     E,(IX+64D) ;LOAD CHAR ON NEXT LINE
39A9  CD083A            CALL   MPY8
39AC  19                ADD    HL,DE
39AD  1E04              LD     E,4D      ;LOOP FOR 2 VERT DOTS
39AF  0608      T2LP1   LD     B,8D      ;LOOP FOR 8 HORZ DOTS
39B1  7E                LD     A,(HL)
39B2  FD21943A          LD     IY,WKCHR
39B6  CB7F      T2LP2   BIT    7,A
39B8  2803              JR     Z,T2CNT-$

39BA  CD433A            CALL   SETBIT
39BD  CB07      T2CNT   RLC    A
39BF  FD23              INC    IY
39C1  10F3              DJNZ   T2LP2-$
39C3  23                INC    HL
39C4  1C                INC    E
39C5  7B                LD     A,E
39C6  FE06              CP     6D
39C8  20E5              JR     NZ,T2LP1-$
39CA  CD203A            CALL   CHROUT  ;OUTPUT WORK CHARS
39CD  C1                POP    BC      ;RESTORE
39CE  E1                POP    HL
39CF  C9                RET
                ;
                ;       SUBROUTINE WHICH PRINTS BOTTOM 6 DOTS OF A CHAR.
                ;
39D0  E5        B6      PUSH   HL        ;WE DESTROY
39D1  C5                PUSH   BC
39D2  CD153A            CALL   INITCHR
39D5  2100F8            LD     HL,CHRDEF
39D8  1600              LD     D,0D
39DA  DD5E40            LD     E,(IX+64D)
39DD  CD083A            CALL   MPY8
39E0  7B                LD     A,E
39E1  C602              ADD    A,2D
39E3  5F                LD     E,A
39E4  19                ADD    HL,DE     ;POINT TO CHAR'S REPRESENTATION
39E5  1E00              LD     E,0D      ;LOOP FOR 6 VERT DOTS
39E7  0608      B6LP1   LD     B,8D      ;LOOP FOR 8 HORZ DOTS
39E9  7E                LD     A,(HL)
39EA  FD21943A          LD     IY,WKCHR
39EE  CB7F      B6LP2   BIT    7,A
39F0  2803              JR     Z,B6CNT-$
39F2  CD433A            CALL   SETBIT
39F5  CB07      B6CNT   RLC    A
39F7  FD23              INC    IY
39F9  10F3              DJNZ   B6LP2-$
39FB  23                INC    HL
39FC  1C                INC    E
39FD  7B                LD     A,E
39FE  FE06              CP     6D
3A00  20E5              JR     NZ,B6LP1-$
3A02  CD203A            CALL   CHROUT  ;OUTPUT WORK CHARS
3A05  C1                POP    BC
3A06  E1                POP    HL
3A07  C9                RET
                ;
                ;       SUBROUTINE WHICH MULTIPLIES DE REG PAIR BY 8
                ;
3A08  CB23      MPY8    SLA    E
3A0A  CB12              RL     D
3A0C  CB23              SLA    E
3A0E  CB12              RL     D
3A10  CB23              SLA    E
3A12  CB12              RL     D
3A14  C9                RET
                ;
                ;       SUBROUTINE WHICH ZEROS WORK CHARS
                ;
3A15  21943A    INITCHR LD     HL,WKCHR

3A18  0608              LD     B,8D
3A1A  3600      INITLP  LD     (HL),0H
3A1C  23                INC    HL
3A1D  10FB              DJNZ   INITLP-$
3A1F  C9                RET
                ;
                ;       SUBROUTINE WHICH WRITES WORK CHARS TO
                ;       PRINTER
                ;
3A20  FD21943A  CHROUT  LD     IY,WKCHR
3A24  0608              LD     B,8D
3A26  FD7E00    CHRLP   LD     A,(IY+0D)
3A29  CD7D3A            CALL   CENOUT
3A2C  FE03              CP     ETX
3A2E  2003              JR     NZ,CHRCNT-$
3A30  CD7D3A            CALL   CENOUT
3A33  FD23      CHRCNT  INC    IY
```

```
3A35  10EF                 DJNZ    CHRLP-$
3A37  C9                   RET
                    ;
                    ;      SUBROUTINE WHICH POSITIONS PRINT HEAD TO
                    ;      START OF NEW LINE
                    ;
3A38  3E03          NEWLN  LD      A,ETX
3A3A  CD7D3A               CALL    CENOUT
3A3D  3E0B                 LD      A,VT
3A3F  CD7D3A               CALL    CENOUT
3A42  C9                   RET
                    ;
                    ;      SUBROUTINE WHICH SETS A BIT (SELECTED BY
                    ;      VALUE IN REG E) IN WORK AREA POINTED TO
                    ;      BY IY
                    ;
3A43  F5            SETBIT PUSH    AF        ;WE DESTROY
3A44  7B                   LD      A,E
3A45  FE00                 CP      0D
3A47  2006                 JR      NZ,CHECK1-$
3A49  FDCB00C6             SET     0,(IY+0D)
3A4D  182C                 JR      SETRTN-$
3A4F  FE01          CHECK1 CP      1D
3A51  2006                 JR      NZ,CHECK2-$
3A53  FDCB00CE             SET     1,(IY+0D)
3A57  1822                 JR      SETRTN-$
3A59  FE02          CHECK2 CP      2D
3A5B  2006                 JR      NZ,CHECK3-$
3A5D  FDCB00D6             SET     2,(IY+0D)
3A61  1818                 JR      SETRTN-$
3A63  FE03          CHECK3 CP      3D
3A65  2006                 JR      NZ,CHECK4-$
3A67  FDCB00DE             SET     3,(IY+0D)
3A6B  180E                 JR      SETRTN-$
3A6D  FE04          CHECK4 CP      4D
3A6F  2006                 JR      NZ,SET5-$
3A71  FDCB00E6             SET     4,(IY+0D)
3A75  1804                 JR      SETRTN-$
3A77  FDCB00EE      SET5   SET     5,(IY+0D)
3A7B  F1            SETRTN POP     AF        ;RESTORE
3A7C  C9                   RET
                    ;
                    ;      SUBROUTINE THAT OUTPUTS CHAR IN REG A

                    ;      TO PRINTER
                    ;
3A7D  F5            CENOUT PUSH    AF
3A7E  F5                   PUSH    AF
3A7F  DBFF          CENBSY IN      A,(0FFH)
3A81  CB7F                 BIT     7,A
3A83  20FA                 JR      NZ,CENBSY-$
3A85  F1                   POP     AF
3A86  F680                 OR      80H
3A88  D3FF                 OUT     (0FFH),A
3A8A  E67F                 AND     7FH
3A8C  D3FF                 OUT     (0FFH),A
3A8E  F680                 OR      80H
3A90  D3FF                 OUT     (0FFH),A
3A92  F1                   POP     AF
3A93  C9                   RET
                    ;
                    ;      WORK AREAS
                    ;
3A94  00            WKCHR  DEFD    00H
3A95  00                   DEFB    00H
3A96  00                   DEFB    00H
3A97  00                   DEFB    00H
3A98  00                   DEFB    00H
3A99  00                   DEFB    00H
3A9A  00                   DEFB    00H
3A9B  00                   DEFB    00H
ERRORS=0000
```

```
B2CNT      3933 B2LP1      3925 B2LP2      392C
B2T4       390E B4CNT      3994 B4LP1      3986
B4LP2      398D B4T2       396F B6         39D0
B6CNT      39F5 B6LP1      39E7 B6LP2      39EE
CENBSY     3A7F CENOUT     3A7D CHECK1     3A4F
CHECK2     3A59 CHECK3     3A63 CHECK4     3A6D
CHRCNT     3A33 CHRDEF     F800 CHRLP      3A26
CHROUT     3A20 DC1        0011 DC3        0013
ETX        0003 FF         000C INITCH     3A15
INITLP     3A1A MLOOP      3872 MPY8       3A08
NEWLN      3A38 PLINE      3899 PLOOP1     389F
PLOOP2     38AE PLOOP3     38C0 PLOOP4     38CE
RS         001E SCREEN     F080 SET5       3A77
SETBIT     3A43 SETRTN     3A7B SPRINT     3856
STX        0002 T2CNT      39BD T2LP1      39AF
T2LP2      39B6 T4CNT      395C T4LP1      394E
T4LP2      3955 T6         38DA T6CNT      38FB
T6LP1      38ED T6LP2      38F4 VT         000B
WKCHR      3A94
```

Dithering Heights

Ken Recters

Is that me on TV? It sort of resembles me, but the image is just black and white with no grays. I walk a few feet further and there I am again, this time with grays, but the motion is jerky. Kind of like seeing every 10th frame of a movie.

At the Midwest Computer Show, a goodly crowd was frequently found around the Computer Station booth. People were watching those crazy images of themselves. They certainly weren't the familiar closed-circuit images produced by a camera and monitor. Something else had been added—an Apple computer. Elbowing closer to the booth, and making way for those who, with screen dumps grasped in their hands, were leaving, I got my first look at the Dithertizer II. I was impressed.

Taking input through a black and white Sanyo VC 1610X camera, the Dithertizer could put a picture on the Apple hi-res screen in less than a second. The person demonstrating the system explained that most of the existing low-cost digitizers take almost five seconds to produce a picture. The Dithertizer requires only 1/60th of a second to grab a binary picture. (The Micro Works system requires almost 5 seconds to grab an image.) The Dithertizer is fast by virtue of the fact that it uses a DMA (direct memory access) type of binary video digitizing versus a frame grabber. These binary frames are combined into dithered matrices of 2X2, 4X4, or 8X8. A 2X2 matrix requires 4 frames, a 4X4 requires 16, and an 8X8 requires 64. Increasing the size of the matrix allows for higher contrast and, therefore, more shades of gray. In other words, an 8X8 matrix provides 64 levels of gray from white to black. The actual picture takes from 1/15 of a second for the smallest matrix to 1 and 1/15 seconds for the largest matrix. Since the image could be redrawn at this rate, a slightly-less-than-real-time motion could be seen on the screen. The term "dithering" refers to the process of producing the appearance of gray scales by means of overlaying.

From what I could see, the user is given many ways to interact with the system. Intensity and contrast are controlled with the paddles. With one press of a key, the Dithertizer can switch to producing straight black and white images with no gray scales. Using the paddles, a variety of interesting contour pictures can be produced. A contour image is produced by subtracting one frame from another so that you end up with just the outline of the image, ie., you just pick up the contour edging. This requires two frames and takes 1/30th of a second to produce. Using contours, images can be produced faster than those with gray scales; you nearly have animation. Using these techniques alone or together allows the user to experiment with various artistic interpretations, or whatever else suits his fancy. Another key freezes the image. From here, it can be saved to disk. The image can also be sent to any printer capable of dumping the hi-res screen. This requires another program which is included with the package.

So, what can you do with the Dithertizer? You can make pictures of your friends, or of anything else a video camera can spy out. With a printer, the pictures could be turned into posters. And, since the scene can be saved from disk and recalled, it can be decorated later with text or shapes from a table. With the right software, many interesting things can be done to the image. With pictures on file, you can spice up your software with great graphics. According to Lynn Sullivan, president of the Computer Station, manufacturers of the Dithertizer, many "real world" applications are possible. For example, gray levels can be extracted from aerial photographs. One professor is using it for work in geology. Another is using it to study sexual response and behavior in rats. It is useful for this because it is a much more sensitive movement detector than a regular video camera.

The Dithertizer II board, with software, sells for $300. The Sanyo VC 1610X camera is $410. If you buy both, the price drops to $650. This $650 package also includes software to produce a hard-copy image on either the Paper Tiger 440 or 460 printers. An Apple Silent Writer printer can also be used by storing the hi-res page and then dumping it out. All the software commands are one letter, for example, C for contour, D for dither and P for print. The package is available from Peripherals Plus, 39 East Hanover Ave., Morris Plains, New Jersey 07950. For further information, call (201) 267-4558.

Lynn Sullivan, president of the Computer Station, as seen by the Dithertizer II.

For other articles on digitizing, see "Periphicon 511 Optical Image Digitizer," *Creative Computing*, vol. 5, number 10. This review presents background information and suggests several interesting applications for computers with cameras. "Image Processing with COMPIC's Computer Portrait System," from vol. 5, number 8, presents a detailed look at the joys and woes of a computer-oriented business, as well as a complete description of the materials required for computer portraits. □

Suddenly, RCA makes talking to your computer a lot cheaper.

New interactive data terminal with color graphics–only $369.*

RCA's new VP-3301 is a professional quality, ASCII encoded, interactive data terminal, suitable for a wide variety of industrial, educational, business and individual applications requiring interactive communication between computer and user. Connects directly to your computer or to a standard modem for over the phone access to time sharing networks and data bases. And it's compatible with networks such as those provided by CompuServe Information Services and Source Telecomputing Corp. Microprocessor intelligence and LSI video control integrated circuits bring performance, features and flexibility at a low price. Power supply included.

Unique color locking circuitry creates sharp, jitter free, true color graphics and rainbow free characters.

Displays the entire field of characters in any of 8 colors against any of 8 background colors (7 gray scales with monochrome monitors). Or to add special emphasis, you can display individual letters, words or lines in different colors or in reverse video.

The VP-3301 offers you a choice of two software-selectable display formats: Either 40 characters by 24 lines. Or 20 characters by 12 lines.

The terminal's resident character set consists of 52 upper and lower case alphabetics, 10 numerals, 32 punctuation/math symbols, and 31 control characters. You can also define a total of 128 of your own characters. Including: Greek letters and other foreign alphabets, graphic symbols, large graphics building blocks, playing card suits, unique character fonts, and "little green men."

The keyboard section features flexible-membrane key switches with contact life rated at greater than five million operations. A finger positioning overlay and positive keypress action give good operator "feel".

An on-board sound generator and speaker provides aural feedback for key presses and may also be activated with escape sequences to provide an audio output.

The sealed keyboard surface is spill proof and dust proof. This combined with high noise immunity CMOS circuitry makes the VP-3301 ideal for hostile environments.

Output is industry standard asynchronous RS232C or 20 mA current loop with six switch selectable baud rates and 8 selectable data formats.

The terminal can be connected directly to a 525 line color or monochrome monitor. Or to a standard TV set using an Rf modulator.

For more information, contact RCA MicroComputer Marketing, New Holland Avenue, Lancaster, PA 17604.

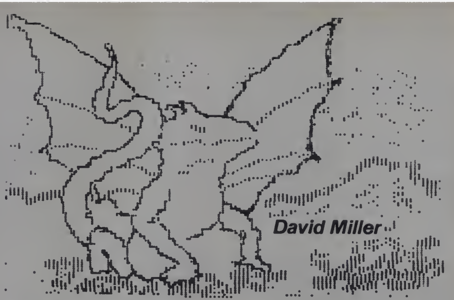
Or call our toll-free number: 800-233-0094.

*Suggested user price

RCA

CIRCLE 245 ON READER SERVICE CARD

Apple-Sketch



David Miller

When I first bought my Apple II computer a few months ago, I was very excited about using the high resolution graphics that would surely dazzle my friends. I had visions of swooping spacecraft catapulting across the screen, of finely detailed game displays; but most of all, I relished the possibility of drawing freely, in color, on the high resolution screen. What possibilities, I thought: a 45000 point display (give or take a few), with six available colors. Surely there would be an easy way to make it do my bidding! So I spent a few evenings working with my shiny new Apple, a color T.V. set, and the friendly Apple manuals, and quickly learned that things might not be very easy after all.

Apple's high resolution graphics mode consists of two areas or "pages" of memory in which to store the information that makes up a high resolution picture. The primary page picture buffer (or page 1) begins at memory location 8192 and extends up to location 16383, while the secondary page immediately follows, from locations 16384 to 24575. Six colors (black, white, green, violet, orange and blue) are available, although some limitations exist that I will talk more about later. Page one is mixture of graphics and text with four text lines residing beneath the graphics screen, although they can be removed by a simple POKE command. Page two is strictly graphics, with no text window easily available to the user. Basic statements can turn points on or off in color and can draw lines between a specified series of points. For those gamers in the crowd (like myself), binary shape tables can be created to draw, rotate, and expand a user defined shape anywhere on the screen. Sounds great, doesn't it? But let's backtrack for a moment.

To put it frankly, a single point plotted on the screen does not a picture make! A lot of careful thought and planning has to come before you can produce eye-catching drawings using this one-at-a-time method. Since many, many points must be used to produce a colorful drawing of, say, your friendly neighborhood computer, it becomes impractical (and very slow) to use a series of HPLOT statements to draw it from within a program. Who'd want to figure out where all those points should go, anyway? The method used to create shape tables is very tedious and difficult; a single mistake can put you right back where you started from.

As a result, many would-be graphics artists have become frustrated with high-resolution graphics and shy away from using them in their programs. The more affluent, of course, buy expensive graphics tablets costing hundreds of dollars, and produce visual effects to make the rest of us turn green with envy. I soon decided to even up the score a little bit; perhaps through a little software magic, it would become possible to duplicate some of the features of the graphics tablets.

The method used to create shape tables is very tedious and difficult; a single mistake can put you right back where you started from.

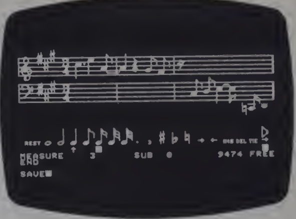
The result is my program, written in Applesoft to make life a little more enjoyable for the artistic Apple owner. With it you can create vivid computer art, without spending hundreds of dollars on an expensive graphics tablet, by using the Apple paddles for the drawing input. All six colors are available for either the actual drawing or for a change of background color; three pen sizes will produce thick strokes or fine detail. If you'd like to throw in some straight lines between any points on the screen, fine; and it's simple to selectively erase parts of a drawing, or the whole screen, instantly. None of this is going to do you much good if you can't preserve the results of your painstaking labor, so it is possible to save drawings on a disk and load them back into memory another time for viewing or modification. What happens if you get stuck and can't remember how to erase the screen, or change color? Type control-H for help! and a page listing all the commands appears in place of the graphics screen; after perusing for a while, hit a key and the graphics screen is instantly restored, with any drawings intact.

When the program is run, a title page appears, followed by the command page after a key is pressed. As I said, you can pop right back here when in trouble by typing control-H. It's very helpful for beginners, but as you grow more experienced you won't be needing it anymore. Press another key and the real fun starts.

The text screen is now replaced by the black high resolution graphics screen, with a flashing white dot hiding somewhere about. If you doubt my word, turn the paddle knobs and it will magically dance across the screen. This is the point mode, as the first text line informs you, and pen position is indicated by that little blinking dot; it's color is white on a black screen and black on the others to improve visibility. To pop back into it when drawing, press P and the point will reappear. You can use it to carefully erase or chip away at parts of a drawing, or to jump quickly from place to place without drawing.

Ready to start a picture? Hit the W key and the dot stops flashing; by manipulating the paddle, you can produce a fine line of small white dots. Quickly move the pen to separate the dots, or draw slowly and carefully for a smooth white line. You can instantly change colors by pressing the first letter of the color desired (control-B for blue to differentiate it from B for black). It's just as easy to change pen size; press S and numerically choose small, medium, or large by pressing 1, 2 or 3. The point mode resumes, so press a color to begin drawing again in the new size. The pen will now be a small box instead of a single point, and a thicker line will result as you draw. Varying pen size within a drawing

David Miller, 79 Hawley Ave., Port Chester, NY 10573.

ALF/Apple Music Synthesizer

With the ALF software, entry of music is easy, fast and accurate.

The ALF Apple Music Synthesizer (AMS) is an easy to use peripheral which allows you to program music into an Apple II computer using standard musical notation. The ALF kit includes the synthesizer board (plugs into any peripheral slot), exceptional quality software, and an extensive user manual.

Sophisticated Music Entry Program

Sheet music is easily entered using the Apple game paddles. The high-resolution ENTRY program features the familiar music staff with a "menu" of musical items listed beneath it (note lengths, rests, edit commands, accidentals, etc.). One game paddle moves a cursor up and down the music staff and is used to select the note pitch; the second paddle chooses from the menu items (note length, etc.) With the ALF hi-res ENTRY program, you won't have to use cryptic codes to select note parameters.

As you program sheet music with ENTRY, measure bars are inserted automatically (and note values are tied over the bar where necessary). Key signatures are also automatic—you don't have to keep writing in every sharp or flat!

Three monophonic, individual parts can be programmed with each ALF Music Synthesizer. Two boards are required for stereo. A total of three synthesizers can be used simultaneously for a maximum of nine voices. By controling the envelope (or shape) of each voice, many different instrumental sounds can be simulated.

Eight-octave Range

The ALF Music Synthesizer has a pitch range of eight octaves—a wider range than a grand piano. The ALF can also play semitones—"blues notes" or the pitches in between the keyboard notes of a piano. (The pitch range is from 27.5 to 55,000 Hertz, well beyond the limits of human hearing.) Tuning accurancy is virtually perfect within two cents of pitch value.

Every parameter of the ENTRY program can be changed again and again during a musical piece. For example, you can make changes in key, time signature, volume, and timbre (envelope). Parts can be edited at any time, also. Notes can be added or deleted, note length can be changed, as well as pitch, volume, etc.

You can save songs on either cassette or disk, and play them back using either ENTRY or PLAY. The playback speed is adjusted with one of the game paddles, and can be varied during the playback, if you wish to change the overall tempo.

Colorful Playback Display

The ALF Music Synthesizer features a 16-color low-res graphic display during song playback. Each musical part is represented on a stylized piano "keyboard"—the intensity of the note determines the color, and the pitch is shown in relation to "middle C".

The ALF Music Synthesizer requires the use of an external audio amplifier. Stereo programming is possible with the use of two or three synthesizer boards.

The ALF software includes the ENTRY and PLAY programs, sample songs, an introduction to "envelope shaping", and demonstrations of advanced uses of the synthesizer.

Nine Voices for only $198

The new ALF "AM-II" music synthesizer offers an unbeatable value for the Apple owner who is a music hobbyist. With nine voices on a single music board for $198.00, the AM-II is the most economical device for creating music with the Apple.

The AM-II uses the same excellent ENTRY and PLAY programs as the more sophisticated ALF Music Synthesizer (AMS); the same hi-res graphic display from which notes are selected with the Apple game paddles (not typed with cryptic codes). All of the conveniences of the ENTRY program apply—easy editing, playback with low-res display, ability to save songs on cassette or disk, etc.

The AM-II has **stereo output** (3 voices in left, 3 voices in the middle, 3 voices in the right).

How can the AM-II offer so much for only $198.00? The two basic differences between the AM-II and the ALF Apple Music Synthesizer (AMS) are pitch accuracy and dynamic range. The AM-II has an accurate pitch range of about six octaves. Pitch values above the treble staff become increasingly inaccurate. Also, the AM-II has a dynamic range of 28db, with 16 different volume levels, (the AMS has a dynamic range of 78db).

The AM-II is manufactured with the same high quality standards as other products from the ALF Corporation. No sacrifice has been made in reliability; the new AM-II is simply a great bargain.

Professional musicians will still want to use the original Apple Music Synthesizer (AMS) for its extended range and volume controls (the AMS has a range of 8 octaves). But for the Apple owner who is interested in music as a hobby, the AM-II is the best music peripheral value available today.

Requires: 16K Apple II or Apple II Plus, cassette or Disk II, and an external audio amplifier (all necessary patch cords are included).

AM-II ALF/Apple Synthesizer	$198.00
AMS ALF/Apple Synthesizer	268.00

To order, send payment plus $3.00 shipping and handling to Peripherals Plus, 119 Maple Ave., Morristown, NJ 07960. Credit card customers should include card number and expiration date of Visa, MasterCard or American Express. Credit card customers may also order toll-free:

800-631-8112

(In NJ call 201-540-0445)

Peripherals Plus

119 Maple Ave., Morristown, NJ 07960

Apple Sketch, continued...

will produce more pleasing results, so use all three to spice up your work. The largest size, by the way, will quickly color in large areas of a picture, such as the sky.

But suppose you'd rather draw black on a white background, or blue on an orange background? No problem! Press X and then the number of the new background color you want; the screen will be instantly filled with that color, and you'll be returned to the point mode. In the process, everything on the screen is erased, so if you hit X by mistake, "return" will get you back to the point mode, safe and sound!

I'm sure that the first few attempts at a drawing will result in a meaningless clutter on the screen; it takes a lot of practice to draw easily with the paddles. By typing control-W, you can erase the whole screen instantly and start over again for (hopefully!) better results. Or you can selectively erase parts of a drawing, either by using the point or with the special erase mode. Press E and a large, blinking square will appear in place of the pen; move it carefully around to erase more quickly than with the point. The eraser is easier to locate than the point, being larger and more visible, so it could also be called a kind of extended point mode.

The line drawing mode can be used to quickly draw straight, even lines between any two points on the screen; a line is drawn in the last pen color used, and its thickness corresponds to the current pen size. After L is pressed, the flashing point reappears on the screen, and its changing X-Y coordinates are continuously displayed in the text window. To set the first endpoint, hit a key; the point will be plotted, followed by a brief pause so that the pen can be moved away. Then the flashing dot reappears, and the second endpoint can be set in the same way. The program asks if there are any corrections, and if the endpoints are exactly where you want them, depress N to draw the line and return to the point mode. Pressing Y gives the option of changing either or both of the endpoints; hit 1 to change point #1, 2 to change point #2, or B to change both. Hitting "return" when in the line drawing mode returns you to the point mode.

Okay, so now you have a great picture done that you want to show your family and friends; how can it be saved? Type control-S and enter the name you'd like it to be stored under; this name should begin with a letter, not a number or control character, although these can be used after the first letter. Hit "return", and the contents of the entire high resolution screen will be saved in a binary file on the disk. Be careful, because an unlocked file with the same name will be erased and written over! If you change your mind about saving, hit "return" to reenter the point mode. The same process holds true for the load command (control-L), but instead of saving, a high resolution picture will be loaded into memory from the specified disk file, gradually filling the graphics screen and erasing anything previously displayed. One important note here: if you load a picture with a colored background, make sure you change the screen to the correct color first. Otherwise, the point mode will start erasing the background, as it might previously have been set for a black screen.

The final command is control-E, and it allows for a graceful, dignified exit from the program, after giving you one last chance to reconsider. More violent types will, I'm sure, use control-C or even (ugh!) reset; make your choice accordingly. And now, for the program itself:

Line 5 resets LOMEM: to 16384, above the primary page of high resolution graphics. While I was entering the program, I painfully discovered that as it grew, it extended upward above location 8192 and into the primary high resolution page, causing the random lines to appear on the graphics screen. The last few lines of the program also acquired bad habits, playing hide and seek with me when more was added to the program. All very unpleasant, but the first line magically cures the problem. Incidentally, LOMEM: is reset to its normal value (just above 2000 with firmware Applesoft), by NEW, DEL, and by adding or changing a program line.

You can instantly change colors by pressing the first letter of the color desired.

Lines 10-50 print the title page, wait for a key to be pressed, clear the keyboard strobe, and go to subroutine 1000, which prints out the command page.

Line 90 traps for an error that might occur during the program run; without it, the program would bomb out, and you would probably lose your picture. It sends program flow to subroutine 4000, where the error is appropriately processed.

Line 100 is initialization; it sets point and screen color, pen size, drawing modes, and D$ as Control-D for DOS commands. It clears the text screen and initializes the high resolution display; then line 105 pops the cursor down to the 21st line directly below the graphics screen and goes to subroutine 650, which handles the point mode and pen size display.

Lines 100-200 handle the drawing, depending on the present program mode and pen size. If a key is pressed, line 110 goes to subroutine 300, which handles commands. Lines 120 and 130 read the two paddles for the X and Y pen position. If the point mode is true, line 140 plots at the present position, erases, resets the color and returns to line 110. Lines 150 and 160 draw a medium and large box in the present color, depending on pen size, if the erase mode is off. If the eraser is on, lines 170 and 180 set its color, depending on the present screen color, and 190 draws and erases it. If none of the above are true, line 200 plots the point and returns to line 110.

The following are program subroutines that are used when commands are input from the keyboard:

Lines 300-385 process commands, calling other subroutines in the process and finally returning to the main drawing section. Line 300 gets the character pressed from the keyboard and clears the keyboard strobe. The remaining lines change colors or modes, or call other subroutines if necessary, depending on the command that has been entered. If the key pressed is not a command, line 385 returns to the main drawing section.

Lines 500-530 save a picture on disk; line 520 alerts DOS with previously initialized D$ and BSAVEs the entire memory contents of high resolution page 1 under the file name input in line 510. A$2000 specifies the starting hexadecimal address (8192 decimal) and L$2000 specifies the length of the area of memory to be stored. Lines 600-630 are used in the same way to load a picture from disk into memory starting at hex location 2000, the beginning of the primary page.

Lines 650-660 set the color of the point for the point mode, depending on the present screen color. Line 660 clears the text window, sets and prints the point mode, goes to subroutine 3000 where the pen size is displayed, and returns.

Lines 700-730 are reached by control-E, and provide a dignified exit from the program. As I said before, killjoys may use "reset" or control-C if they wish.

Lines 800-840 print out the numbers for the different colors and get a character response. "Return" or CHR$ (13) pops you back into the drawing routine after resetting the point mode; an incorrect response causes line 815 to get another character. Otherwise, line 830 sets the new screen color, plots a point, and calls the special machine language routine 62454 to clear the screen in the last color HPLOTed. Finally, line 840 goes to subroutine 650, which handles the point mode restoration and pen size display.

Lines 900-940 use the same method to get a new pen size; a valid response resets

Apple Sketch, continued...

```
PRINT "80      "
]LIST
5  LOMEM  16384
10  HOME   VTAB 10. HTAB 7  PRINT "***     APPLE-SKETCH      ***". PRINT
20  HTAB 13  PRINT "BY DAVID MILLER"
30  VTAB 19· HTAB 11  PRINT "HIT ANY KEY TO BEGIN"
40  IF  PEEK ( - 16384) <  = 127 THEN 40
50  POKE  - 16368,0  GOSUB 1000
90  ONERR  GOTO 4000
100  HOME   HGR .OLDCLR = 3: HCOLOR= 3.SCRNCLR = 0·PNT = 1.ER = 0 SML = 1.
     MED = 0 LRG = 0.D$ =  CHR$ (4)
105  VTAB 21. GOSUB 650
110  IF  PEEK ( - 16384) > 127 THEN  GOSUB 300
120 X =  PDL (1) * 1.21 + 2  IF X > 277 THEN X = 277
130 Y =  PDL (0) + 2  IF Y > 157 THEN Y = 157
140  IF PNT THEN  HPLOT X,Y  HCOLOR= SCNRCLR  HPLOT X,Y. HCOLOR= OLDCLR: GOTO
     110
150  IF MED AND ER = 0 THEN  HPLOT X,Y: HPLOT X - 1,Y - 1 TO X + 1,Y - 1 TO
     X + 1,Y + 1 TO X - 1,Y + 1 TO X - 1,Y - 1· GOTO 110
160  IF LRG AND ER = 0 THEN  HPLOT X,Y: HPLOT X - 1,Y - 1 TO X + 1,Y - 1 TO
     X + 1,Y + 1 TO X - 1,Y + 1 TO X - 1,Y - 1  HPLOT X - 2,Y - 2 TO X + 2
     ,Y - 2 TO X + 2,Y + 2 TO X - 2,Y + 2 TO X - 2,Y - 2: GOTO 110
170  IF ER AND SCNRCLR = 0 THEN OLDCLR = 3
180  IF ER AND SCRNCLR <  > 0 THEN OLDCLR = 0
190  IF ER THEN  HCOLOR= OLDCLR  HPLOT X,Y. HPLOT X - 1,Y - 1 TO X + 1,Y -
     1 TO X + 1,Y + 1 TO X - 1,Y + 1 TO X - 1,Y - 1  HCOLOR= SCRNCLR  HPLOT
     X,Y: HPLOT X - 1,Y - 1 TO X + 1,Y - 1 TO X + 1,Y + 1 TO X - 1,Y + 1 TO
     X - 1,Y - 1  GOTO 110
200  HPLOT X,Y. GOTO 110
299  REM  *** COMMAND SUBROUTINE
300  GET A$  POKE  - 16368,0
305  IF A$ =  CHR$ (23) THEN  HCOLOR= SCRNCLR. HPLOT 0,0. CALL 62454: HCOLOR=
     OLDCLR  RETURN
310  IF A$ = "W" THEN  HOME   VTAB 21  PRINT "COLOR=WHITE":OLDCLR = 3: HCOLOR=
     3·PNT = 0:ER = 0. GOSUB 3000  RETURN
315  IF A$ =  CHR$ (2) THEN  HOME   VTAB 21  PRINT "COLOR=BLUE":OLDCLR = 6
      HCOLOR= 6.PNT = 0·ER = 0  GOSUB 3000. RETURN
320  IF A$ = "G" THEN  HOME   VTAB 21. PRINT "COLOR=GREEN":OLDCLR = 1. HCOLOR=
     1:PNT = 0 ER = 0  GOSUB 3000  RETURN
325  IF A$ = "E" THEN  HOME   VTAB 21: PRINT "ERASE MODE" PNT = 0 ER = 1  RETURN

330  IF A$ = "V" THEN  HOME   VTAB 21  PRINT "COLOR=VIOLET":OLDCLR = 2  HCOLOR=
     2 PNT = 0.ER = 0  GOSUB 3000  RETURN
335  IF A$ = "O" THEN  HOME   VTAB 21  PRINT "COLOR=ORANGE".OLDCLR = 5: HCOLOR=
     5·PNT = 0 ER = 0  GOSUB 3000  RETURN
340  IF A$ = "B" THEN  HOME   VTAB 21  PRINT "COLOR=BLACK".OLDCLR = 0. HCOLOR=
     0 PNT = 0 ER = 0  GOSUB 3000  RETURN
345  IF A$ =  CHR$ (12) THEN  HOME   VTAB 21  GOSUB 600. RETURN
350  IF A$ = "S" THEN  HOME   VTAB 21  GOSUB 900. RETURN
355  IF A$ = "P" THEN  HOME   VTAB 21  GOSUB 650  RETURN
360  IF A$ =  CHR$ (0) THEN  GOSUB 1000  GOSUB 650  RETURN
365  IF A$ =  CHR$ (19) THEN  HOME   VTAB 21· GOSUB 500  RETURN
370  IF A$ = "L" THEN  GOSUB 2000  RETURN
375  IF A$ = "X" THEN  GOSUB 800
380  IF A$ =  CHR$ (5) THEN  GOSUB 700
385  RETURN
500  REM   *** SAVE PICTURE
510  PRINT "READY TO SAVE PICTURE    ",  INVERSE   PRINT "RETURN";  NORMAL
       PRINT " TO EXIT"  INPUT "PICTURE NAME ? ",PICTURE$
515  IF PICTURE$ = "" THEN  GOSUB 650  RETURN
520  PRINT D$,"BSAVE ";PICTURE$,", A$2000, L$2000"
530  PRINT PICTURE$," SAVED. "  FOR I = 1 TO 3000. NEXT I  GOSUB 650: RETURN

600  REM   *** LOAD PICTURE
610  PRINT "READY TO LOAD PICTURE    ";  INVERSE   PRINT "RETURN";· NORMAL
       PRINT " TO EXIT"  INPUT "PICTURE NAME ? ";PICTURE$
615  IF PICTURE$ = "" THEN  GOSUB 650. RETURN
620  PRINT D$,"BLOAD ",PICTURE$,", A$2000"
630  PRINT PICTURE$," LOADED "  FOR I = 1 TO 3000  NEXT I. GOSUB 650  RETURN

649  REM  ***  SET POINT MODE COLOR
650  IF SCNRCLR = 0 THEN OLDCLR = 3. GOTO 660
655 OLDCLR = 0
660  HCOLOR= OLDCLR PNT = 1  HOME   VTAB 21  PRINT "POINT MODE": GOSUB 300
     0· RETURN
669  REM  ***  CTRL-E  EXIT PROGRAM?
700  HOME   VTAB 21  PRINT "REALLY EXIT THE PROGRAM (Y/N) ? ",
705  GET A$
710  IF A$ = "Y" THEN  TEXT   HOME   GOTO 4100
720  IF A$ = "N" THEN  GOSUB 650  RETURN
730  GOTO 705
799  REM  *** CHANGE BACKGROUND COLOR
800  HOME   VTAB 21  PRINT "BLACK-0   GREEN-1   VIOLET-2   WHITE-3"  PRINT
     "ORANGE-5   BLUE-6   ";  INVERSE   PRINT "RETURN";. NORMAL . PRINT "
     TO EXIT"
810  PRINT "NEW BACKGROUND COLOR # ",
815  GET Z$  IF Z$ =  CHR$ (13) THEN 840
820  IF  ASC (Z$) < 48 OR  ASC (Z$) > 54 OR Z$ = "4" THEN 015
830 SCRNCLR =  VAL (Z$)  HCOLOR= SCRNCLR  HPLOT 0,0. CALL 62454  HCOLOR= O
     LDCLR
```

pen size and goes to subroutine 650, which handles point mode and pen size display. An incorrect response causes line 905 to get another character.

Lines 1000-1210 display the command page. Line 1000 puts the screen back into the text mode and clears it; after the commands are printed, line 1200 waits for a key to be pressed. Then 1210 clears the keyboard strobe and uses POKE-16304,0 to restore the high resolution graphics screen without clearing it to black (which is what another HGR command would do).

Lines 2000-2470 handle the line drawing, displaying the X and Y coordinates for each point and HPLOTing the lines. Lines 2060 and 2070 read the paddles for point #1, and 2190 and 2200 read them for point #2.

Lines 3000-3030 check for the current pen size and display it beneath the lower right of the graphics screen.

Finally, lines 4000-4070 handle any DOS errors that might occur when loading or saving picutres. Line 4010 sets A to the Apple error code for the trapped

It takes a lot of practice to draw easily with the paddles.

error, which is stored in decimal location 222; it also sets B equal to the line that the error occurred on. Then it clears the text screen and tabs down to the top of the text window, ready to print a message based on the error. The lines after handle:

1. Attempts to load a picture not on disk (DOS "file not found" error).
2. Attempts to save on a write protected disk, such as the DOS master or one with a write protect tab covered.
3. An I/O error, usually encountered when the disk drive door is left open.
4. Attemts to save on a filled disk, or with too little space left to hold the picture.
5. Attempts to save under a file name on disk that is locked.
6. The use of an illegal file name, usually beginning with a number or a control character.
7. Attempts to load a Basic or text file as a picture (believe me, that just won't work!)
8. Control-C: returns the text mode and ends.

As far as I can see, those are the only errors that would normally occur when running the program; all other input is handled by GET and thrown out if inappropriate. If all else fails, line 4060 tells you what error occurred and where before stopping the program. Before

```
840  GOSUB 650  RETURN
899  REM  *** CHANGE PEN SIZE
900  PRINT "READY TO CHANGE PEN SIZE"· PRINT "SMALL(1) - MEDIUM(2) - LARGE
     (3) ?";
905  GET A$
910  IF A$ = "1" THEN SML = 1.MED = 0.LRG = 0  GOSUB 650. RETURN
920  IF A$ = "2" THEN SML = 0.MED = 1 LRG = 0  GOSUB 650: RETURN
930  IF A$ = "3" THEN SML = 0 MED = 0.LRG = 1  GOSUB 650  RETURN
940  GOTO 905
999  REM  *** COMMAND PAGE
1000  TEXT   HOME   HTAB 6  PRINT "***   LIST OF COMMANDS   ***"  PRINT   PRINT
1010  INVERSE   PRINT "CTRL",  NORMAL   PRINT " B - SETS DRAWING COLOR TO
     BLUE"
1020  PRINT  TAB( 6),"W - SETS DRAWING COLOR TO WHITE"
1030  PRINT  TAB( 6);"B - SETS DRAWING COLOR TO BLACK"
1040  PRINT  TAB( 6),"O - SETS DRAWING COLOR TO ORANGE"
1050  PRINT  TAB( 6),"G - SETS DRAWING COLOR TO GREEN"
1060  PRINT  TAB( 6),"V - SETS DRAWING COLOR TO VIOLET"
1070  PRINT
1080  PRINT  TAB( 6);"P - CHANGES TO POINT MODE"
1090  PRINT  TAB( 6),"E - CHANGES TO ERASE MODE"
1100  PRINT  TAB( 6);"L - CHANGES TO LINE DRAWING MODE"
1110  PRINT  TAB( 6),"X - CHANGES BACKGROUND COLOR"
1120  PRINT  TAB( 6);"S - CHANGES PEN DRAWING SIZE"
1130  PRINT
1140  INVERSE   PRINT "CTRL";  NORMAL   PRINT " W - WIPE SCREEN CLEAR, STA
     RT OVER"
1150  INVERSE   PRINT "CTRL",  NORMAL   PRINT " S - SAVE CURRENT PICTURE O
     N DISK"
1160  INVERSE   PRINT "CTRL";. NORMAL   PRINT " L - LOAD PICTURE FROM DISK
     "
1170  INVERSE   PRINT "CTRL",  NORMAL   PRINT " H - DISPLAY THIS PAGE OF C
     OMMANDS"
1180  INVERSE   PRINT "CTRL",  NORMAL   PRINT " E - EXIT THE PROGRAM"
1190  PRINT   PRINT  TAB( 6);"HIT ANY KEY WHEN READY ."
1200  IF  PEEK ( - 16384) <  = 127 THEN 1200
1210  POKE  - 16368,0  POKE  - 16304,0  RETURN
2000  REM  *** DRAW LINE
2010 LINE$ = "                "
2020 CH = 0·X1 = 0·X2 = 0 Y1 = 0·Y2 = 0 QX = 0 QY = 0
2030  HOME   VTAB 21  PRINT "LINE DRAWING MODE         ",  INVERSE   PRINT
     "RETURN";. NORMAL   PRINT " TO EXIT"
2040  VTAB 22  PRINT "POINT #1  X=     Y=     <HIT KEY TO SET"
2050  IF  PEEK ( - 16384) > 127 THEN  GET A$  POKE  - 16368,0. VTAB 22  HTAB
     25. PRINT LINE$ QX = X1:QY = Y1  GOSUB 2430  GOTO 2150
2060 X1 =  PDL (1) * 1 21 + 2  IF X1 > 277 THEN X1 = 277
2070 Y1 =  PDL (0) + 2. IF Y1 > 157 THEN Y1 = 157
2075  VTAB 22: HTAB 13. PRINT  INT (X1),
2080  IF X1 < 10 THEN  PRINT "   "
2090  IF X1 >  = 10 AND X1 < 100 THEN  PRINT "  "
2100  IF X1 >  = 100 THEN  PRINT " "
2105  VTAB 22. HTAB 20· PRINT  INT (Y1),
2110  IF Y1 < 10 THEN  PRINT "   "
2120  IF Y1 >  = 10 AND Y1 < 100 THEN  PRINT "  "
2130  IF Y1 >  = 100 THEN  PRINT " "
2140  HCOLOR= OLDCLR. HPLOT X1,Y1  HCOLOR= SCRNCLR  HPLOT X1,Y1  GOTO 2050

2150  IF CH = 1 THEN 2290
2160  IF A$ =  CHR$ (13) THEN 2300
2170  VTAB 23  PRINT "POINT #2  X=     Y=     <HIT KEY TO SET"
2180  IF  PEEK ( - 16384) > 127 THEN  GET A$  POKE  - 16368,0  VTAB 23  HTAB
     25: PRINT LINE$ QX = X2 QY = Y2. GOSUB 2430. GOTO 2280
2190 X2 =  PDL (1) * 1 21 + 2· IF X2 > 277 THEN X2 = 277
2200 Y2 =  PDL (0) + 2  IF Y2 > 157 THEN Y2 = 157
2205  VTAB 23· HTAB 13  PRINT  INT (X2),
2210  IF X2 < 10 THEN  PRINT "   "
2220  IF X2 >  = 10 AND X2 < 100 THEN  PRINT "  "
2230  IF X2 >  = 100 THEN  PRINT " "
2235  VTAB 23  HTAB 20  PRINT  INT (Y2),
2240  IF Y2 < 10 THEN  PRINT "   "
```

running the program, I suggest taking a CATALOG of the disk to see what pictures are available, and which are locked (it would be a good idea to lock an important picture to prevent accidental erasure). Since you won't be able to save onto a locked file name, you'd have to save under another name to preserve your present version without losing the program.

Once a picture has been saved on disk, it can be recalled from a Basic program and displayed using the same technique as in subroutine 600. The program should initialize the string D$ as control-D or CHR$(4), then set the primary page of high resolution graphics and load the picture into memory using DOS commands.

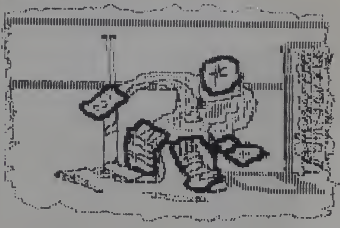
There are a few limitations to what you can do with this program. One problem arises in that the Apple paddles are not very sturdy and tend to become worn with usage. Theoretically, they should produce a steady stream of values from 0 to 255. When new, yes. After a few weeks of Space Invaders and assorted other

If all else fails, line 4060 tells you what error occurred.

paddle games, no. Presently, my PDL(O) only goes up to about 180, and PDL(1) to 225. As a result, I use PDL(O) for the Y input, which only has to go to about 160. Unfortunately, the screen is 280 units wide, so my PDL(1) values have to be multiplied to make the pen cover the entire screen, and in the process, some values are lost. With multiplication, I've discovered that about two out of every ten X values just cannot be produced, and it becomes impossible to draw a perfectly smooth line in the X direction. Is there a solution to the problem?

Yes, even if only a partial one, and I suggest it for those who type in this program. The first step is to find out just how your particular paddles will go. Change lines 2060 and 2070 to X1=PDL(1) and Y1=PDL(O). Then enter the line drawing mode and move your paddles clockwise from zero to the highest possible values. Say your paddles are functioning perfectly, and PDL(1) stops at 255. You *can* use the paddle values without multiplying if you leave a small margin on either side of the page. For a perfect paddle, subtract 255 from 280 and divide by 2, leaving a margin of 12.5 on each side. Let's call it 12, and change line 2060 to read:

```
2060 X1=PDL(1)+12:IF X1> 277
     THEN X1=277
```



```
2250  IF Y2 >  = 10 AND Y2 < 100 THEN  PRINT "  "
2260  IF Y2 >  = 100 THEN  PRINT " "
2270  HCOLOR= OLDCLR  HPLOT X2,Y2  HCOLOR= SCRNCLR  HPLOT X2,Y2: GOTO 2180
2280  IF A$ =  CHR$ (13) THEN 2300
2290  VTAB 21  HTAB 1. PRINT "ANY CORRECTIONS ? ";
2295  GET A$
2300  IF A$ =  CHR$ (13) THEN  HCOLOR= SCRNCLR: HPLOT X1,Y1: HPLOT X2,Y2. HCOLOR
= OLDCLR  GOTO 2460
2310  IF A$ = "Y" THEN 2360
2320  IF A$ = "N" AND SML THEN  HPLOT X1,Y1 TO X2,Y2  GOTO 2450
2330  IF A$ = "N" AND MED THEN  FOR I =  - 1 TO 1  HPLOT X1 + I,Y1 + I TO
   X2 + I,Y2 + I  NEXT I  GOTO 2450
2340  IF A$ = "N" AND LRG THEN  FOR I =  - 2 TO 2  HPLOT X1 + I,Y1 + I TO
   X2 + I,Y2 + I  NEXT I  GOTO 2450
2350  GOTO 2295
2360  VTAB 21: HTAB 1  PRINT LINE$,"   ": VTAB 21  HTAB 1: PRINT "POINT 1,
   2, OR BOTH ?",
2370  GET A$
2380  IF A$ =  CHR$ (13) THEN 2300
2390  IF A$ = "1" THEN CH = 1  GOSUB 2470  HPLOT X1,Y1  HCOLOR= OLDCLR  GOTO
   2040
2400  IF A$ = "2" THEN CH = 2  GOSUB 2470  HPLOT X2,Y2  HCOLOR= OLDCLR. GOTO
   2170
2410  IF A$ = "B" THEN CH = 0  GOSUB 2470  HPLOT X1,Y1. HPLOT X2,Y2: HCOLOR=
   OLDCLR  GOTO 2040
2420  GOTO 2370
2430  IF A$ =  CHR$ (13) THEN  RETURN
2440  HCOLOR= OLDCLR  HPLOT QX,QY  FOR I = 1 TO 500  NEXT I  RETURN
2450  HOME   VTAB 21  PRINT "DONE.  "  FOR I = 1 TO 1000  NEXT I
2460  GOSUB 650  RETURN
2470  VTAB 21  HTAB 1. PRINT LINE$,"     ": VTAB 21: HTAB 1  PRINT "LINE D
   RAWING MODE"  HCOLOR= SCRNCLR. RETURN
2999  REM  *** PEN SIZE DISPLAY
3000  VTAB 21  HTAB 25: PRINT "PEN SIZE=";
3010  IF SML THEN  PRINT "SMALL ": RETURN
3020  IF MED THEN  PRINT "MEDIUM". RETURN
3030  IF LRG THEN  PRINT "LARGE "  RETURN
4000  REM  *** ERROR TRAPPING
4010 A =  PEEK (222):B =  PEEK (218) +  PEEK (219) * 256. HOME : VTAB 21
4020  IF A = 6 THEN  PRINT "YOU DON'T HAVE THAT PICTURE ON DISK...": GOTO
   4070
4025  IF A = 4 THEN  PRINT "YOUR DISK IS WRITE PROTECTED !": PRINT "USE AN
   OTHER OR REMOVE WRITE PROTECT TAB "  GOTO 4070
4030  IF A = 8 THEN  PRINT "I/O ERROR   IS YOUR DISK DRIVE OPEN ?": PRINT
   "IF NOT, TRY AGAIN OR USE ANOTHER DISK. "  GOTO 4070
4035  IF A = 9 THEN  PRINT "TOO MANY FILES ON DISK, DELETE SOME": PRINT "O
   R USE ANOTHER DISK ": GOTO 4070
4040  IF A = 10 THEN  PRINT "THAT DISK FILE IS LOCKED.  .UNLOCK IT": PRINT
   "OR USE ANOTHER NAME FOR YOUR PICTURE. "  GOTO 4070
4045  IF A = 11 THEN  PRINT "ILLEGAL FILE NAME. "  PRINT "PLEASE BEGIN WITH
   A LETTER "  GOTO 4070
4050  IF A = 13 THEN  PRINT "THAT FILE IS NOT A PICTURE.  ."  GOTO 4070
4055  IF A = 255 THEN  TEXT  HOME  GOTO 4100
4060  PRINT "PROGRAM TERMINATED DUE TO ERROR ";A  PRINT "LINE # ";B  GOTO
   4100
4070  FOR I = 1 TO 3000  NEXT I  GOSUB 650  GOTO 110
4100  TEXT  END
```

Now you can get *all* of the X values between 12 and 267, and can draw a perfectly smooth line. You *won't* be able to go outside of those two margins, but they are really very small and I believe the sacrifice is worth it. Likewise, change lines 120 and 2190 to read:

```
120 X=PDL(1)+12:IF X>277 THEN X=277

2190 X2=PDL(1)+12:IF X2>277
     THEN X2=277
```

I've altered my version in the same way; as my paddle goes to 225, I add 27 to the PDL(1) values and lose a somewhat larger margin. It's all up to your personal preference; remember, these are very inexpensive input devices and it's going to take a little experimentation to get them working just right.

I'd also like to warn you that certain colors will not draw cleanly on colored backgrounds, due to the way the high resolution display works. Drawing black on an orange background, for example, produces ragged green fringes, which may also appear when erasing orange on another background. These color fringes appear mainly when using the colored backgrounds. They don't seem to be a problem when drawing on black or white. Consequently, I'd advise drawing on either black or white; if you'd like large areas of another color, switch to the large pen and color them in. Of course, the fringe effects may be just what you want to produce a dazzling, modernistic picture.

The fringe effects may be just what you want to produce a dazzling, modernistic picture.

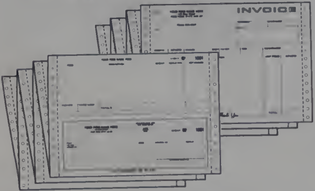
Don't forget, graphics tablets cost money for a good reason: they are very fast, accurate, and sophisticated. A simple Basic program, such as mine, is hard put to match their performance using very inexpensive paddles as input devices. It can, however, provide a creative challenge and hours of plain old fun if the user has just a little patience and self control (please try not to offend or abuse your poor machine in any way). I can't promise miracles, but believe me, you *can* draw good pictures if you use some imagination and creativity. I know, because I've done some myself that I like very much! Inevitably, some people just won't be able to get the knack of drawing with the paddles, no matter how hard they try. Ah well, they can always go back to playing Space Invaders.... □

FREE COMPUTER FORMS KIT

with continuous business forms for small computer systems

Each kit contains samples, programming guides, flyers, prices and order forms for checks, invoices, statements, envelopes, stock paper and labels to fit almost every computer system.

- **Available in quantities of 500, 1,000, 2,000 4,000, 6,000**
- **Low Prices (500 checks only $32.50)**



- **FAST SERVICE — It is our policy to ship within 6 working days following our receipt of your order.**
- **MONEY BACK GUARANTEE — If for any reason you are not completely satisfied, your money will be promptly refunded.**

Fast Service by mail or. . .PHONE TOLL FREE
1+800-225-9550
Mass. residents 1+800-922-8560
8:30 a.m. to 5:00 p.m. Eastern Time Monday — Friday

Please Ship FREE Kit To:	**CODE 460**
Name	
Company	
Street	
City, State and Zip	
Phone	
Computer make & model	

Nebs Computer Forms
78 Hollis Street, Groton, Mass. 01450
A division of New England Business Service Inc

✓291

CIRCLE 167 ON READER SERVICE CARD

BLITZ BUG LIGHTNING & TRANSIENT VOLTAGE SUPPRESSOR

LETTERS OF RECOMMENDATION:

$24.95
DELIVERY FROM STOCK
ADD $1.50 SHIPPING & HANDLING
N.J. RESIDENTS ADD 5% SALES TAX
DEALER INQUIRIES INVITED

OMNI COMMUNICATIONS COMPANY, Inc.
200 WEST COUNTY LINE ROAD
R.D. 3, BOX 200
JACKSON, NEW JERSEY 08527
(201) 928-1477 (609) 259-2617

CIRCLE 174 ON READER SERVICE CARD

Bridge the GAP in your Business Accounting.

GENERAL ACCOUNTING PACKAGE. This is a proven double entry accounting system with user definable accounts. The account numbers are made up of 7 4-digit fields allowing 7 levels of account classifications. With the use of the **Operator Report Selector Generator** (ORSG), you can generate any type of report you desire, or use report programs in GAP-GL, GAP-AP, and GAP-AR.

GAP-GL Includes all basic GAP functions, plus entry of General Ledger transactions, prints General Journal, General Ledger summary and detail, Balance Sheet, Profit and Loss . . . . Price $124.95

GAP-AR Requires GAP-GL to run, allows adding A/R invoices, printing Sales Journal, detail A/R report, Acct. Aging, add/update Cash Receipts with register, Cash Receipts Journal, and A/R Billing . . . . Price $99.95

GAP-AP Requires GAP-GL to run, allows adding of A/P invoices, printing Purchase Journal, detail A/P report, Aging of Accounts, Check Writing, Check Printing, Cash Disbursements Journal . . . . Price $99.95

SAVE NOW by purchasing all three packages for only $299.95. Simply mention this ad when calling in your order, or send the ad with your mail order.

Your **BA/VISA** or **MasterCard** is welcomed.
Call today to receive complete package specifications.
System requirements are 48K CP/M. CP/M is a registered trademark of Digital Research.

PROFESSIONAL DATA SYSTEMS
318 E. 18th Street
Bakersfield, CA 93305
Telephone (805) 323-0891

IWI

CIRCLE 228 ON READER SERVICE CARD

Apple II Lo-Res Shape Tables

David Lubar

Drawing a complex shape in Lo-Res graphics requires a large number of PLOT, HLIN, and VLIN statements. Getting such a shape to move on the screen can be a slow process. While each square is drawn quickly, time is lost since the process involves interpreting the Basic command, jumping to the monitor, returning the Basic, and so on. In such cases, a Lo-Res shape table subroutine could be useful. The following article describes such a program, designed for use with Integer Basic.

I attempted to follow, to a degree, the format of Hi-Res shape tables, while eliminating some of the more difficult aspects of such tables.

What is a Shape Table?

A shape table is just a series of instuctions which are represented as numbers. In this case, the numbers contain two types of information: 1) Whether or not to plot a square at the present location; 2) Where to move next. With this information, any shape can be defined, as long as it fits within the limits of the screen. There are eight possible directions to move in the Lo-Res routine (see figure 1.) Combined with the plotting options, this gives 16 different commands. Since the Apple's monitor uses hexadecimal data, and since there are two hex digits in a byte, each byte can contain two table entries. To further simplify plotting, the table is constructed without an index. This restricts the entire table to 256 bytes, which isn't much of a limitation.

Making a Table

Each table must begin with a \$00 (the "\$" signifies that the number is in hex.) The end of each shape within the table is also marked with a \$00. As in Hi-Res, you can't move up twice without plotting. But, since diagonal moves are allowed, you can get there by going diagonally left and up, then diagonally right and up.

Starting at the top, the values of the moves go clockwise from 0 to 7 (figure 1.) If the point is to be plotted, 8 is added to the value. Once all the values for a shape have been calculated, they are put in pairs. The routine reads each byte from right to left, so the first command of each pair should be the lower digit of the byte. For example, if the first command has a value of "8" and the second a value of "F", the table entry would be "F8". (For those who aren't familiar with hex, the values "10" through "15" are represented as "A" through "F".) Figure 2 illustrates the process of assembling a shape table.

With this information, any shape can be defined, as long as it fits within the limits of the screen.

Using the Program

Since the routine takes values from the variable table, certain variables have to be defined first. This is done with:

10 X0=Y0=SHAPE

Any variables can be used, as long as they are the same length as the ones shown above. Whenever you want to draw a shape from the table, define X0 for the X coordinate, Y0 for the Y coordinate, and SHAPE for the desired shape. The draw is done with CALL 4353. After this, X0 and Y0 will have whatever values the last move assigned to them. To draw the shape elsewhere, X0 and Y0 must be redefined.

The table begins at location \$1000, and can go up to \$10FF. The routine lies directly above this point. That leaves 2K for the Basic variable table. Some refinements could be added to the program, such as error checks to make sure the squares are plotted within screen limits. If desired, a scale function could be added.

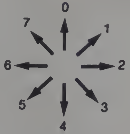


Figure 1.
Plotting Vectors

Direction	No Plot		Plot	
	Binary	Hex	Binary	Hex
↑	0000	0	1000	8
↗	0001	1	1001	9
→	0010	2	1010	A
↘	0011	3	1011	B
↓	0100	4	1100	C
↙	0101	5	1101	D
←	0110	6	1110	E
↖	0111	7	1111	F

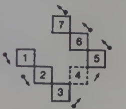
Putting it all Together

The routine can be appended to an Integer Basic program in a number of ways. It can be loaded separately, it can be loaded with the program if the pointers are first reset, or it can be POKEd from Basic as explained in "The Apple Cart" column (*Creative Computing*, March 1980.) The table can also be POKEd from basic, or loaded in together with the routine.

If you are unfamiliar with machine lan-

guage, there is an easy way to enter the routine into memory. Just go into the monitor, type "1100:" followed by the bytes shown in the hex dump (Figure 4.) Only the first memory location has to be entered. After that, whenever you hit RETURN, type another colon before beginning the next row of bytes. Once the whole routine is entered, you can check it against the disassembled listing (Figure 3) by typing "1100L". This will show the first twenty instructions. After this, type "L (RETURN)" for each additional twenty lines.

The listing shown by the Apple will contain the commands that are in the second column of Figure 3. The third column will be represented on the screen as numbers instead of the labels shown.




```
1 = 1011 = B
2 = 1011 = B
3 = 1001 = 9
4 = 0001 = 1
5 = 1111 = F
6 = 1111 = F
7 = 1111 = F
```

Figure 2. Sample Table Entry

Note: For last square plotted, direction of move after plot is arbitrary. Also, if there are an odd number of nybbles, the last one is completed with a 0 in the hi position.

Figure 3

```
* LORES SHAPE TABLE SUBROUTINE
* ENTERED FROM BASIC WITH CALL 4353
XO     EQU  $805
YO     EQU  $80C
SHAPE  EQU  $816
DATA   EQU  $1000
PLOT   EQU  $F800
       ORG  $1100
*
RTN    RTS
START  LDY  SHAPE     ;GET SHAPE NUMBER
       LDX  #$0       ;INITIALIZE COUNTER TO 0
LOOP   LDA  DATA,X    ;GET BYTE FROM SHAPE TABLE
       INX
       CMP  #$0       ;END OF A SHAPE?
       BNE  LOOP      ;NO.  KEEP LOOKING
       DEY            ;YES.  DESIRED SHAPE IS FOUND WHEN
       BNE  LOOP      ;Y EQUALS 0
DRAW   LDA  DATA,X    ;GET BYTE TO BE PLOTTED
       CMP  #$0       ;SHAPE DONE?
       BEQ  RTN       ;YES.  GO BACK TO BASIC
       PHA            ;NO.  SAVE BYTE
       AND  #$F       ;GET LOW NYBBLE
       CMP  #$8       ;PLOT?
       BCC  NEXT      ;NO.  SKIP PLOTTING ROUTINE
       PHA            ;YES.  SAVE LOW NYBBLE
       LDA  YO        ;GET Y COORDINATE
       LDY  XO        ;GET X COORDINATE
       JSR  PLOT      ;MONITOR PLOT ROUTINE
       PLA            ;RESTORE LO NYBBLE
NEXT   JSR  COORD     ;FIND NEW COORDINATES FOR X AND Y
       PLA            ;GET ORIGINAL BYTE
       LSR            ;SHIFT HI NYBBLE LO
       LSR
       LSR
       LSR
       CMP  #$8       ;SAME AS ABOVE
       BCC  NEXT1
       PHA
       LDA  YO
       LDY  XO
       JSR  PLOT
       PLA
NEXT1  JSR  COORD
       INX            ;POINT TO NEXT BYTE
       JMP  DRAW      ;DO IT ALL AGAIN
*
* THE FOLLOWING SECTION HANDLES THE MOVE SET BY THE NYBBLE
*
```

—CONTINUED ON NEXT PAGE—

These can be checked by using the symbol table at the top of Figure 3.

Practical Considerations

If a program uses very simple shapes, this routine isn't necessary. But, when you want to quickly draw and move a complicated shape, such as a person or a large spaceship, you'll find that the Lo-Res Shape Table routine allows much quicker animation than is possible in Basic. It also requires fewer program statements. Finally, as a bonus, it simplifies the creation of shapes. You don't have to worry about coordinates. All you need to know is which direction you want to move and whether you want a square at that location.

When you want to quickly draw and move a complicated shape, such as a person or a large spaceship, you'll find that the Lo-Res Shape Table routine allows much quicker animation than is possible in Basic.

How It Works

This section can be skipped by anyone who prefers to avoid the company of bit hackers and other fanatics.

First, Y is loaded with the number of the shape. The routine steps through the table, decrementing Y whenever a $00 is found. When Y=0, the desired shape has been located. Each byte of the table is put in the A register, and a check is made to see if the shape is done. If not, the byte is pushed onto the stack. An AND #F gets the lo nybble. If the hi bit of the nybble is set to 1, the square will be plotted. The A register is pushed onto the stack again since the monitor PLOT routine destroys this register. The X and Y coordinates are taken from the variable table and placed in the Y and A registers. After the monitor PLOT, the nybble is pulled from the stack. An AND #$7 reduces it to the three-bit move value. The move is accomplished by incrementing or decrementing X0 and

Lo-Res, continued...

```
COORD  AND   #$7       ;REDUCE TO 3 BIT VALUE
       CMP   #$0       ;FIND DIRECTION VALUE
       BEQ   UP
       CMP   #$1
       BEQ   DIAG1
       CMP   #$2
       BEQ   RIGHT
       CMP   #$3
       BEQ   DIAG2
       CMP   #$4
       BEQ   DOWN
       CMP   #$5
       BEQ   DIAG3
       CMP   #$6
       BEQ   LEFT
DIAG4  DEC   XO        ;IF IT REACHES HERE, A=7
       DEC   YO
       RTS
UP     DEC   YO
       RTS
DIAG1  INC   XO
       DEC   YO
       RTS
RIGHT  INC   XO
       RTS
DIAG2  INC   XO
       INC   YO
       RTS
DOWN   INC   YO
       RTS
DIAG3  DEC   XO
       INC   YO
       RTS
LEFT   DEC   XO
       RTS
```

Figure 3

Y0 as necessary. Next, the stack is pulled again, getting the original byte. Four LSR's put the hi nybble into the lo position. Then the PLOT (if the hi bit of the nybble is set) and move are done. X is incremented to point to the next byte, and the routine loops back to draw.

```
1100- C0 AC 1E 08 02 00 BD 00
1108- 10 F8 C9 00 D0 F8 88 D0
1110- F5 BD 00 10 C9 00 F0 F8
1118- 48 29 0F C9 08 30 0B 48
1120- AD 0C 08 AC 05 08 20 00
1128- F8 68 20 46 11 68 4A 4A
1130- 4A 4A C9 08 90 0B 48 AD
1138- 0C 08 AC 05 08 20 00 F8
1140- 68 20 48 11 F0 4C 11 11
1148- 29 07 C9 00 F0 1F C9 01
1150- F0 1F C9 02 F0 22 C9 03
1158- F0 22 C9 04 F0 25 C9 05
1160- F0 25 C9 06 F0 28 CE 05
1168- 08 CE 0C 08 60 CE 0C 08
1170- 60 FF 05 08 CE 0C 08 60
1178- EE 05 08 60 EE 05 08 EE
1180- 0C 08 60 FE 0C 08 60 CE
1188- 05 08 FF 0C 08 60 CE 05
1190- 08 60
```

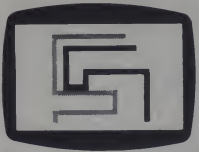
Figure 4

If LOMEM isn't set to $800, the values of X0, Y0, and SHAPE in the symbol definitions will have to be adjusted. To relocate the program, just change the value of DATA and the value of JMP DRAW.

That's all there is to it, unless, as mentioned before, you want to add error checks (including one to make sure that SHAPE isn't given a value for which there is no table entry). Happy plotting. □

Strategy Games

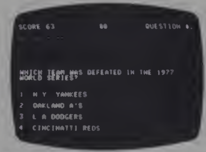
Cassette CS-4003 $11.95 4 Programs Requires 16K Apple II or Apple II Plus



Blockade. Build a wall to trap your opponent, but don't hit anything.

UFO. Use lasers, warheads or guns to destroy an enemy spacecraft.

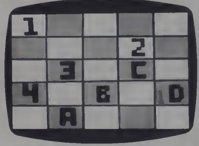
Skunk. A 2-player strategy game played with dice, skill and luck.



Genius. A fast-moving trivia quiz with scores of questions.

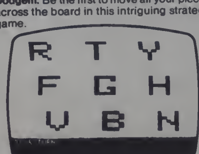
Brain Games

Cassette CS-4004 $11.95 7 programs Requires 16K Apple II or Apple II Plus

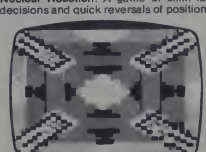


Dodgem. Be the first to move all your pieces across the board in this intriguing strategy game.

Nuclear Reaction. A game of skill, fast decisions and quick reversals of position.



Parrot. A Simon-type game with letters and tones. **Dueling digits** is a version with numbers.



Midpoints and Lines. Two colorful graphics demonstrations. **Tones** lets you make music and sound effects.

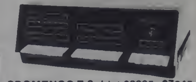
Strategy and Brain Games are also available on one 32K Apple disk (CS-4502) for $24.95. Add $2.00 shipping per order. Send to **Creative Computing Software**, P.O Box 789-M, Morristown, NJ 07960. Or call 800-631-8112.

CIRCLE 300 ON READER SERVICE CARD

Buy By Mail and Save!

COMPUTERS

INTERTEC SuperBank, 32K	$2495
64K Ram, List $3345	$2695
64K Quad, List $3995	$3395
NORTH STAR Horizon I	
32K DD List $2695	$1989
Horizon I QD List $2995	$2245
Horizon 2 32K DD, List $3095	$2289
IntersystemDP-1 List $1749	$1495



CROMENCO Z-2, List $9995	$7945
System 64K, List $3990	$3179
System 3 64K, List $7395	5689
ATARI 800, List $1080	$799
APPLE II, 16K	$969

DISK SYSTEMS

THINKER TOYS' Discus 2D	$939
Dual Discus 2D	$1559
Discus 2 + 2, List $1549	$1259
M26 Hard Disk, List $4995	$3949

PRINTERS & TERMINALS

PAPER TIGERS IDS-440	$679
With graphic option	$749
CENTRONICS 730-1, List $795	$595
737, List $995	$789
704-9 180 cps	$1495
703-9 180 cps	$1569
T I 810, List $1895	$1489
NEC SPINWRITER5530	$2395
NEC SPINWRITER 5515	$2395
DIABLO 630 List $2711	$2399
INTERTEC	
Intertube III, List $895	729
Emulator	$729
Televideo 912C	$679
920C	$799
Hazeltine 1420	$789
1500	$845
Soroc 120, List $995	$689
Soroc 140	$994

Most items in stock for immediate delivery. Factory sealed cartons, w/full factory warranty. NYS residents add appropriate sales tax. Prices do not include shipping. VISA and Master Charge add 3%. C.O.D. orders require 25% deposit. Prices subject to change without notice.

Computers Wholesale

P.O. Box 144 Camillus, NY 13031

(315) 472-2582

CIRCLE 132 ON READER SERVICE CARD

Hi-Res Text For The Apple

Paul Hitchcock

If this hasn't happened to you yet, beware—

You're sitting in front of your Apple II Plus, the graphics master and number-cruncher *extraordinaire*. Behind you a group of friends impatiently await the unveiling of your latest *Meisterwerk*: The graphical solution of Schroedinger's wave equation. The quantum universe begins to unfurl across the screen of your monitor; you, however, secretly listen for the inevitable murmurs of astonished approval. And when the last hi-res dot winks on the screen, you turn to the audience for the expected-but-highly-deserved applause. But instead of the roar of the crowd, you are confronted with a roomful of knitted brows. Time holds its breath while your confidence ebbs away. Suddenly a question rips apart the icy silence. No, not a question, but a searing, air-ionizing laser blast aimed to demolish the foundations of your programming expertise:

"Well, *surely* it labels the axes, doesn't it?"

A red haze diffuses across your eyes; through the blur you see your friends filing silently out of the room. Your barely audible mutterings (But Applesoft doesn't include a hi-res character set...) tip-toe across the room to fall on ears that will not hear. You have just been control-C'd into the Twilight Zone.

Maybe the preceding anecdote is a *little* exaggerated, but it does emphasize an important point: a graphics display should convey all of the information a user needs to understand what the display means. That generally means graphics *and* text. Without a generous sprinkling of alphanumerics, most graphs and charts and games are, in three words, boring, dull, and boring. But even more to the point, "naked" graphics are uninformative. A quick glance at the two histograms in Figure 1 will show you precisely what I mean. Although the subject of "swimsuit sales" may not move you to the edge of your chair, at least you know what the graph is trying to say.

So how do you obtain a hi-res character set for your Apple? One inexpensive answer is found in the Apple's ability to draw user-defined, high-resolution "shapes." Listing 1 defines a table of such shapes which will give you the entire alphabet, ten digits, as well as several special characters (See Figure 2). By using the Applesoft DRAW, XDRAW, SCALE, and ROT commmands in conjunction with this table, you will be able to print text quickly and easily on the hi-res screen.

I said the shape table was inexpensive, but it's not completely free: it will cost you 641 bytes of RAM. But I think you'll agree the price is reasonable when you see how

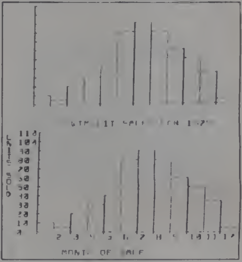


Figure 1

```
*1000.1280

1000- 38 00 78 00 7A 00 7C 00
1008- 7E 00 80 00 82 00 90 00
1010- 92 00 94 00 9C 00 A4 00
1018- A6 00 AE 00 B0 00 B5 00
1020- B9 00 C1 00 D0 00 D8 00
1028- E4 00 EF 00 F9 00 05 01
1030- 11 01 18 01 28 01 32 01
1038- 34 01 36 01 3F 01 48 01
1040- 51 01 53 01 55 01 62 01
1048- 6F 01 79 01 84 01 90 01
1050- 99 01 A4 01 B1 01 BA 01
1058- C2 01 CF 01 D7 01 E3 01
1060- F1 01 FC 01 07 02 15 02
1068- 22 02 2D 02 37 02 43 02
1070- 4F 02 5C 02 68 02 75 02
1078- 01 00 01 00 01 00 01 00
1080- 01 00 25 25 0E 18 2C BE
1088- 09 3E 3E 66 49 3E 2C 00
1090- 01 00 01 00 24 2D 9F 32
1098- 36 2D 07 00 24 3F 8D 32
10A0- 36 3F 05 00 01 00 34 0E
10A8- 2D 36 24 2D 07 00 01 00
10B0- 98 29 2D 07 00 12 37 25
10B8- 00 25 25 9E 33 37 37 04
10C0- 00 25 25 3F 3F 36 36 25
10C8- E5 93 2E 2D 25 24 04 00
10D0- 24 37 0A 36 3E 2D 07 00
10D8- 25 25 3F 3F 4E 32 3F 36
10E0- 2D 2D 07 00 2D 24 3F 3F
10E8- 96 32 2D 2D 24 04 00 2D
10F0- 24 0E C3 36 2D 2D 36 26
10F8- 00 3F 24 2D 2D DE 2A 35
1100- 36 3F 3F 04 00 35 35 3E
1108- 3F 27 2C 0C 9F 24 2C 2D
1110- 00 25 25 3F 3F 56 31 37
1118- 37 3D 00 2D 24 3F 3F 36
1120- AD 2D 36 3F 3F 24 2D 00
1128- 2D 24 3F 3F 36 6D 31 36
1130- 04 00 01 00 01 00 2C FD
1138- 3A 3E 35 35 2D 07 00 2D
1140- 97 3D 3F 2F 08 38 2F 00
1148- 3C 6F 2A 2E 37 37 3F 05
1150- 00 01 00 01 00 2D 36 E6
1158- D8 26 24 ED 23 25 2D 2E
1160- 06 00 2E 2E 3E 3F 27 24
1168- 24 2D 35 35 3F 3F 00 89
1170- 32 3F 3F 24 24 2C 2D 35
1178- 00 29 3E 3E 3E 27 24 24
1180- 2D 2E 06 00 AD 12 3F 3F
1188- 24 2C DD 24 2D 2D 07 00
1190- 45 38 3F 37 2E DD 36 26
1198- 00 2A 35 3E 3F 27 24 24
11A0- 2D 2D 06 00 2D 24 0E 18
11A8- 34 2E DD 36 66 49 26 24
11B0- 00 24 2F 3D 36 36 3E 2D
11B8- 07 00 09 24 36 36 3E 3F
11C0- 27 00 2C 25 0E C3 36 36
11C8- 66 51 21 27 27 3F 00 52

11D0- 31 3F 3F 24 24 34 00 3C
11D8- 3C 36 36 66 49 26 24 24
11E0- 37 07 00 2E 36 25 24 24
11E8- DE 33 3D 24 37 36 36 04
11F0- 00 09 24 3F 3F 36 36 2E
11F8- 2D 25 24 00 38 35 2D 2C
1200- 3C 3C 3F 36 36 26 00 09
1208- 24 3F 3F 36 36 2E 2D 8C
1210- AD 27 25 24 04 2E 2E E6
1218- D8 26 24 24 2D 35 35 3F
1220- 3F 00 BF 12 2D 2D 24 3C
1228- DF 24 2D 2D 00 24 3F 4E
1230- 09 3C 97 36 26 00 00 09
1238- 24 1E D8 33 36 36 2D 2D
1240- 24 04 00 09 24 1E 08 33
1248- 36 35 2E 25 2C 04 00 36
1250- 35 25 24 24 1E 08 33 36
1258- 36 25 05 00 25 25 3E D8
1260- 33 35 35 37 37 2C 2C 35
1268- 35 04 00 2D 24 1E D8 33
1270- 2E 35 36 04 00 25 25 3F
1278- 3F 95 31 37 37 2D 2D 07
1280- 00
```

Listing 1

Paul Hitchcock, 2309 Blake St., #308, Berkeley, CA 94704.

much sparkle a bit of text will lend to your graphics displays.

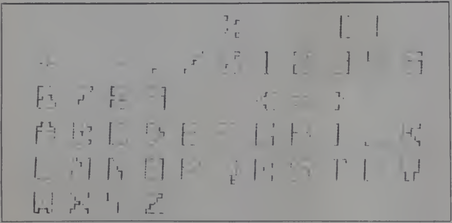


Figure 2

Entering the Shape Table

To load the shape table, power up your Apple and enter the monitor by typing "CALL-151 (return)". Enter each line of the table as shown in Listing 1, except replace the "-" after each line number with a colon. After entering the table, you must store its length and its starting address in locations $0-$1 and $E8-$E9, respectively. You can accomplish these latter two operations with the following monitor commands:

∅:8∅ ∅2 (return), and
E8:∅∅ 1∅ (return).

When you have finished the above, go back to Applesoft with CTRL C (return). Type in and RUN the short test program given in Listing 2. If your television or monitor display looks substantially like Figure 2, you're ready to save the table on tape. If it doesn't, go back and fix things up. Unless you're the type who remembers phone numbers and zip codes in hex, it's all too easy to make a mistake.

Listing 2

```
10  HGR : HCOLOR= 3
20  SCALE= 3
30  ROT= 0
40 N = 1
50  FOR J = 10 TO 135 STEP 25
60  FOR I = 7 TO 257 STEP 25
70  IF N = 60 THEN 120
80  DRAW N AT I,J
90 N = N + 1
100  NEXT I
110  NEXT J
120  END
```

Saving the Table on Tape

To save the table on tape, re-enter the monitor (CALL-151) and type

∅.1W 1∅∅∅.1280W

Put a clean tape in your recorder, start recording, and press (return). After two beeps the cursor will reappear, indicating the table has been saved.

For your own mental well-being, it would be a good idea to make three or four copies of the shape table. I'm sure you know what I mean: you make several copies of something on tape, and each copy loads perfectly. Just make a single copy, though, and it will invariably fail to load. While Mr. Murphy might chuckle with smug satisfaction, this kind of thing turns me into a hapless psychotic.

In the future, you enter the shape table into your computer just as you would enter a cassette program, with one exception: instead of typing "LOAD", you type "SHLOAD". You can load the shape table even if you already have a program in memory, so SHLOAD can actually be activated as a program statement. However, those of you with Apples having only 16K RAM must take a special precaution before loading your tables. When you enter the table via the SHLOAD command, the Apple places the table immediately below the memory address specified by the current HIMEM setting. But when the Apple is first turned on, HIMEM is set to the highest avaialable memory address–16384, if you have a 16K machine. When you enter the hi-res graphics mode with the command HGR, all of the addresses from 8192 to 16384 (the hi-res picture bufer) are cleared, i.e., set to zero. So you must set HIMEM to 8192 (or lower) before you load your table, to avoid automatically erasing it.

Text Writing in Hi-Res

The shapes in the table are numbered, and to draw a particular shape on the screen, you refer to its number in a DRAW statement. For example, the letter "A" has the number 34, and you draw this letter at the screen coordinates X, Y with the command "DRAW 34 AT X, Y". Now the fifty-nine shapes in the table have been arranged so that the table number of a given shape is related to the ASCII number of the character the shape represents through the following formula:

N(character)=ASC("character")-31,

where N is the table number of the character enclosed in parentheses. With this function, you can forget about table numbers and can draw a character just by referring to the character itself. Again using the letter "A" as an example, the command "DRAW ASC("A")-31 AT 3∅, 5∅" will cause the letter "A" to be drawn at the screen coordinates 3∅, 5∅.

By using the above formula in conjunction with Applesoft's built-in string functions, you can easily write text strings in hi-res; Listing 3 defines a subroutine I have used in several programs for just this purpose. Before you call the subroutine though, you have to initialize four variables:

XN,YN—the coordinates of the first letter of the string

N$—the string to be printed

ZN—the horizontal/vertical printing flag. When the

flag is down (ZN= ∅), printing proceeds horizontally;

when it is up (ZN= ∅), the string will be printed in

the vertical (downward) direction.

Listing 4 provides a short example of how to use the text printing subroutine and Figure 3 shows the screen output of the program in Listing 4.

I said the shape table was inexpensive, but it's not completely free: it will cost you 641 bytes of RAM.

Advice and Limitations

You've probably noticed that I failed to include a number of special characters in the shape table (!/"/#/$/&/'/:/;/?/@). Your programs won't bomb if you include any of these characters in a text string—the illegal character will just be interpreted as a space character. The decision to omit these characters was completely arbitrary; if you want to add some (or all) of them to your table, you'll find the instructions for

```
300  HGR : HCOLOR= 3
310  POKE - 16302,0: REM  FULL SCREEN
320 ZN = 0: REM  HORIZONTAL PRINTING
330 N$ = "APPLE SINE"
340 XN = 105:YN = 20
350  GOSUB 20000
360  FOR XN = 6 TO 273 STEP 7
370 YN = 79 - 40 *  SIN (6.28 * (XN - 6)
 / 139)
380 ZN = 1: REM  VERTICAL PRINTING
390  GOSUB 20000
400  NEXT XN
410  HPLOT 2,0 TO 279,0 TO 279,191 TO 2,
191 TO 2,0
420  END
```

Listing 4

doing so in the *Applesoft II Reference Manual.* Should you need only a limited number of the shape table characters for a particular application, the manual will also show you how to "cannibalize" the table to get the characters you want.

The shape table character set does suffer from one minor functional limitation: you may only print white (HCOLOR= 3) text on a black (HCOLOR= Ø) background, or vice versa. Because of the way the Apple displays colored lines in hi-res, an attempt to write text using any other color combination will result in missing line segments in all of the printed characters. In short, whatever you write will be unintelligible, although it might look pretty stylish.

As far as letter and line spacing is concerned, I've found the following rule-of-thumb to be useful: the distance between two characters or two lines of characters should be seven times the value of the SCALE; that is, if you print the first character of a text string at the coordinates X,Y, and you have previously specified that SCALE= S, the next character of the string will have the coordinates X+7*S,Y. By the way, if you follow this rule with SCALE= 1, you have room for twenty-three lines of hi-res text with forty characters per line.

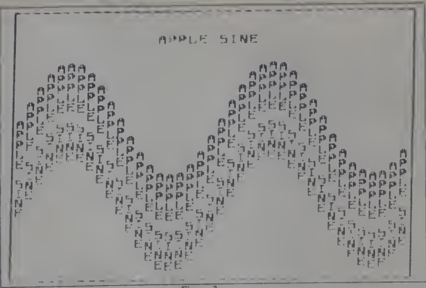


Figure 3

```
20000  REM *HI-RES PRINTING SUBROUTINE*
20005  SCALE= 1: ROT= 0
20010  FOR CN = 1 TO  LEN (N$)
20020 O$ =  MID$ (N$,CN)
20030  IF ZN THEN 20060
20040  DRAW  ASC (O$) - 31 AT XN + 7 * (
CN - 1),YN
20050  GOTO 20070
20060  DRAW  ASC (O$) - 31 AT XN,YN + 7
* (CN - 1)
20070  NEXT CN
20080 ZN = 0
20090  RETURN
```

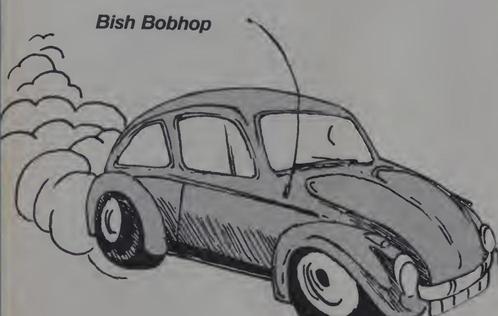
Listing 3

The subroutine in Listing 3 represents only an elementary example of what you will be able to do with the character set. By experimenting with different letter and line spacings, you should be able to output hi-res text in nearly any conceivable format. Your graphs and your games will begin to communicate much more effectively—and isn't that what graphics is all about?

I hope you find this simple character set to be as useful as I've found it to be. So—as far as hi-res text is concerned—write soon, and write often! □

Lit'l Red Bug

Bish Bobhop



Lit'l Red Bug is a game of skill in which you must drive your Volkswagen along a road without going off the pavement. The faster you go, the more points you get...unless you leave the road. Off the road, you begin to lose points. The faster you go, the more points you lose!

If you don't have a color display, you might want to change the last part of line 10 to read: CAR=15 instead of: CAR=1.

This will change the color of the "bug" to white so that it will show up better on a black and white set.

The program is written in Apple-II integer Basic with just a touch of machine language for class.

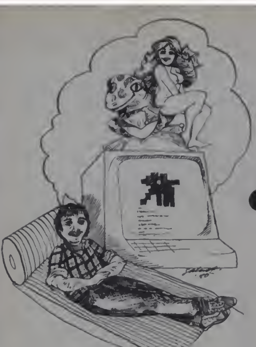
Bish Bobhop, Apple Computer, 10260 Brandley Dr., Cupertino, CA 95014.

Note: Besides the Basic portion of *Lit'l Red bug*, there are three portions of machine code which have to be entered; two programs and one data table. The easiest way to enter this code is through direct hex entry in the monitor. For the table, just type 1000: followed by the data. If RETURN is used, begin the next line with another colon. Once this table is entered, the two short programs can be entered in two ways. Either the assembler can be used, or more simply, the hex code in the second column can be entered. For each program, enter the starting address, followed by a colon, then start typing the groups of numbers in the second column. Again, when you hit return, type another colon. After entering all the code, check the results by typing the first address followed by an L. This should cause the screen to display exactly what is shown in the listing. To save the data on cassette, use 1000.1163W. To save it on disk, use BSAVE TITLE, A$1000, L$164.

Second note: The author, Bish Bobhop, is the lesser-known brother of a certain mythical figure who works deep in the halls of the Elppa Computer Company.

```
*1000.103F

1000-  00 80 00 80 00 80 00 80
1008-  28 A8 28 A8 28 A8 28 A8
1010-  50 D0 50 D0 50 D0 50 D0
1018-  00 00 00 00 00 00 00 00
1020-  04 04 05 05 06 06 07 07
1028-  04 04 05 05 06 06 07 07
1030-  04 04 05 05 06 06 07 07
1038-  00 0A 0A 0A 0A 08 0B 0A

10AE-   A2 13       LDX   #$13
10B0-   BD 00 10    LDA   $1000,X
10B3-   85 02       STA   $02
10B5-   BD 20 10    LDA   $1020,X
10B8-   85 03       STA   $03
10BA-   CA          DEX
10BB-   BD 00 10    LDA   $1000,X
10BE-   85 00       STA   $00
10C0-   BD 20 10    LDA   $1020,X
10C3-   85 01       STA   $01
10C5-   A0 27       LDY   #$27
10C7-   B1 00       LDA   ($00),Y
10C9-   91 02       STA   ($02),Y
10CB-   88          DEY
10CC-   10 F9       BPL   $10C7
10CE-   8D FF FF    STA   $FFFF
10D1-   E0 00       CPX   #$00
10D3-   D0 DB       BNE   $10B0
10D5-   60          RTS
10D6-   00          BRK

10EE-   A5 00       LDA   $00
10F0-   85 02       STA   $02
10F2-   68          PLA
10F3-   48          PHA
10F4-   C6 02       DEC   $02
10F6-   D0 FA       BNE   $10F2
10F8-   8D 30 C0    STA   $C030
10FB-   C6 01       DEC   $01
10FD-   D0 EF       BNE   $10EE
10FF-   60          RTS
1100-   A9 90       LDA   #$90
1102-   85 00       STA   $00
1104-   A9 40       LDA   #$40
1106-   85 01       STA   $01
1108-   20 EE 10    JSR   $10EE
110B-   A9 74       LDA   #$74
110D-   85 00       STA   $00
110F-   A9 57       LDA   #$57
1111-   85 01       STA   $01
1113-   20 EE 10    JSR   $10EE
1116-   A9 50       LDA   #$50
1118-   85 00       STA   $00
111A-   A9 61       LDA   #$61
111C-   85 01       STA   $01
111E-   20 EE 10    JSR   $10EE
1121-   A9 4D       LDA   #$4D
1123-   85 00       STA   $00
1125-   A9 00       LDA   #$00
1127-   85 01       STA   $01
1129-   20 EE 10    JSR   $10EE
112C-   A9 5D       LDA   #$5D
112E-   85 00       STA   $00
1130-   A9 61       LDA   #$61
1132-   85 01       STA   $01
1134-   20 EE 10    JSR   $10EE
1137-   A9 4D       LDA   #$4D
1139-   85 00       STA   $00
113B-   A9 00       LDA   #$00
113D-   85 01       STA   $01
113F-   20 EE 10    JSR   $10EE
1142-   A9 4D       LDA   #$4D
1144-   85 00       STA   $00
1146-   A5 00       LDA   $00
1148-   85 01       STA   $01
114A-   20 EE 10    JSR   $10EE
114D-   A9 4D       LDA   #$4D
114F-   85 00       STA   $00
1151-   A9 00       LDA   #$00
1153-   85 01       STA   $01
1155-   20 EE 10    JSR   $10EE
1158-   A9 4D       LDA   #$4D
115A-   85 00       STA   $00
115C-   A5 00       LDA   $00
115E-   85 01       STA   $01
1160-   4C EE 10    JMP   $10EE
```

```
5 X=Y=L=DL=R=DR=SPEED=POS=TOT=TIME=Z=BACK
10 ROAD=5:FIELD=12:BUSH=4:CAR=1
20 DOWN=4096+174
40 GOSUB 2000
60 TOT=0:TIME=425:Y=TIME/12+3
70 POS=1500:X=POS/75:BACK=ROAD
80 L=16:DL=0:R=20:DR=0
100 COLOR=BACK: VLIN Y-1,Y AT X
150 CALL DOWN
160 Z= PEEK (-16336):Z= PEEK (-16336)
200 POS=POS+ PDL (0)-128: IF POS<0 THEN POS=0: IF POS>2999
    THEN POS=2999

220 TIME=TIME-1:Y=TIME/12+3
230 X=POS/75:BACK= SCRN(X,Y)
250 COLOR=CAR: VLIN Y-1,Y AT X
300 COLOR=FIELD: HLIN 0,39 AT 1: HLIN 0,39 AT 0
310 COLOR=BUSH: PLOT RND (40),1
320 COLOR=ROAD: HLIN L,L+4 AT 1: HLIN R,R+4 AT 1: HLIN L,L+4
    AT 0: HLIN  R,R+4 AT 0
350 Z= PEEK (-16336):Z= PEEK (-16336)
400 IF NOT RND (5) THEN DL= RND (3)-1
410 IF NOT RND (5) THEN DR= RND (3)-1
420 IF L=1 THEN DL=1: IF L=33 THEN DL=-1
430 IF R=1 THEN DR=1: IF R=33 THEN DR=-1
450 L=L+DL:R=R+DR
480 SPEED=(255- PDL (1))/5
490 FOR K=0 TO SPEED: NEXT K
500 TOT=TOT+60-SPEED
520 IF BACK=FIELD THEN 1000
540 Z= PEEK (-16336):Z= PEEK (-16336)
550 VTAB 22: TAB 9: PRINT TIME/6;" ";: TAB 22: PRINT TOT;
    "  ";: TAB 37: PRINT 50-SPEED
600 IF TIME>0 THEN 100
700 TEXT : CALL -936: VTAB 5
710 TAB 12: PRINT "--- GAME OVER ---"
720 VTAB 12: PRINT "  YOUR SCORE OF ";TOT;" IS ";
730 IF TOT>10000 THEN 750
740 PRINT "TERRIBLE!": PRINT : IF TOT<5000 THEN PRINT "(YOU
    SHOULD BE ASHAMED OF YOURSELF!)"
750 IF TOT>10000 AND TOT<=15000 THEN PRINT "PRETTY BAD!"
760 IF TOT>15000 AND TOT<=19000 THEN PRINT "ABOUT AVERAGE."
770 IF TOT>19000 AND TOT<=22000 THEN PRINT "VERY GOOD!"
780 IF TOT>22000 THEN PRINT "EXCELLENT!"
790 IF TOT>24000 THEN CALL 4352
800 FOR K=1 TO 1000: NEXT K
810 VTAB 20: PRINT "PUSH EITHER PADDLE BUTTON TO PLAY AGAIN"
820 IF PEEK (-16286)<128 AND PEEK (-16287)<128 THEN 820
850 GOSUB 2050
900 GOTO 60
1000 FOR K=1 TO 20:Z= PEEK (-16336): NEXT K
1010 TOT=TOT-5*(50-SPEED):TIME=TIME-1
1020 IF TOT<0 THEN TOT=0
1050 GOTO 550
2000 TEXT : CALL -936
2010 VTAB 10: TAB 10: PRINT "*** LIT'L RED BUG ***"
2020 PRINT : TAB 19: PRINT "BY"
2030 PRINT : TAB 15: PRINT "BISH BOBHOP"
2040 FOR K=1 TO 1500: NEXT K
2045 GOSUB 3000
2050 GR : POKE -16298,0
2060 VTAB 22: PRINT "  TIME:";: TAB 15: PRINT "SCORE:";:
     TAB 30: PRINT "SPEED:
     "
2070 POKE 50,63: VTAB 24: TAB 2: PRINT "  *** LIT'L RED BUG ***
     ";: POKE 50,255
2100 FOR K=0 TO 39
2110 COLOR=FIELD: HLIN 0,39 AT K
2120 COLOR=BUSH: IF K MOD 2 THEN PLOT RND (40),K
2130 COLOR=ROAD: HLIN 16-K/8,24+K/8 AT K
2140 NEXT K
2200 RETURN
3000 CALL -936: TAB 10: PRINT "*** LIT'L RED BUG ***"
3010 PRINT : PRINT : PRINT "THE OBJECT OF THIS GAME IS TO
     DRIVE YOUR"
3020 PRINT "LITTLE RED VOLKSWAGEN DOWN THE ROAD AS"
3030 PRINT : PRINT "FAST AS YOU CAN WITHOUT GOING OFF THE"
3040 PRINT : PRINT "PAVEMENT.  GAME PADDLE 0 IS YOUR STEER-"
3050 PRINT : PRINT "ING WHEEL, AND PADDLE 1 IS YOUR ACCELER-"
3060 PRINT "ATOR.  HAVE A NICE RIDE!"
3090 FOR K=1 TO 1000: NEXT K
3100 VTAB 22: PRINT "PUSH EITHER GAME PADDLE BUTTON TO BEGIN"
3110 IF PEEK (-16286)<128 AND PEEK (-16287)<128 THEN 3110
3120 IF PEEK (-16286)>127 OR PEEK (-16287)>127 THEN 3120
3200 CALL -936: RETURN
```



The Digital Couch

David Tunbo

The Digital Couch program turns your computer into a psychiatrist. It draws a picture not unlike an inkblot and gives three choices as to what the picture looks like. The program keeps score and rates the player when the "tests" are over.

The program draws a picture by POKEing a graphics block symbol at the center of the screen. One of four directions is selected at random and another block is added after checking that the new block will not cross a previous one, or go off of the screen. When it becomes trapped and cannot place a new block, the picture is finished.

The pictures match the titles more often than you might think.

Lines 400-412 are the choices for the picture titles, and are selected at random. The list can be expanded to your heart's (and patience's) desire. If the player picks the correct answer (also chosen at random) then one is added to their score, and a new picture is drawn. If the player's choice is wrong, then a randomly selected subroutine (lines 500-900) asks some nosy questions and makes a few disparaging remarks. These can also be expanded.

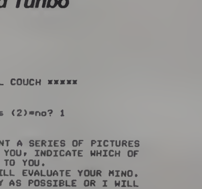
When the "tests" are over the player is "graded" on how well he or she has performed, and a diagnosis is reached.

I use the program at parties, and the results are very entertaining. The drawing progression is fascinating to watch, and the drawings sometimes become quite complex. The pictures match the titles more often than you might think.

The Digital Couch was written on an Ohio Scientific Challenger II. □

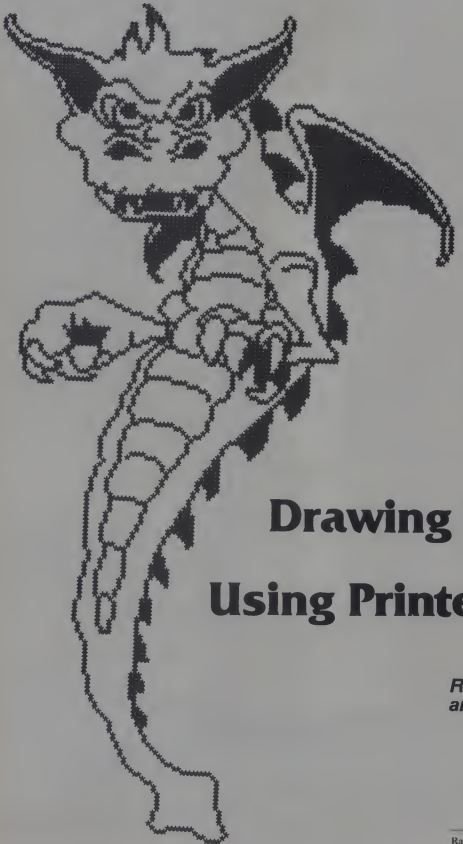
David Tunbo, 747 N. Neblett, Stephenville, TX 76401.

```
RUN
                ***** THE OIGITAL COUCH *****


Do you want instructions (1)=yes (2)=no? 1

THE OOCTOR WILL SEE YOU NOW.
     HELLO.  I'M GOING TO PRESENT A SERIES OF PICTURES
FOR YOU TO LOOK AT.  WHEN I ASK YOU, INDICATE WHICH OF
THE THREE CHOICES IT LOOKS LIKE TO YOU.
     BASED ON YOUR ANSWERS, I WILL EVALUATE YOUR MINO.
ANSWER THE QUESTIONS AS HONESTLY AS POSSIBLE OR I WILL
CALL YOUR MOMMY ON YOU.



How many tests do you want? 1
Hit 'return' for test # 1 ?
```



```
IS THIS A:
( 1 ) RODNEY
( 2 ) CREEPY CANAL
( 3 ) PEANUT BUTTER JET PLANE
? 3
NO, IT'S A RODNEY
YOUR ANSWER IS NOT CORRECT.....BUT IT IS MEANINGFUL.


LET'S SEE NOW.
YOUR SCORE IS  0 %

I WOULD SAY...
YOU REALLY ARE A MESS!!

OK
LIST

OK
LIST

 5 OIM H$(12),F$(3),L$(6):S=0
 6 Z=0:W=0
 8 GOSUB 400
 9 FOR T=1 TO 30:PRINT:NEXT T
 10 PRINTTAB(15)'***** THE OIGITAL COUCH *****'
 20 PRINT:PRINT:INPUT'Do you want instructions (1)=yes (2)=no';B
 30 IF B=1 THEN 1000
 40 IF B=2 THEN 50
 45 GOTO20
 50 PRINT:PRINT:INPUT'How many tests do you want';C
 52 IF C<=0 THEN 50
 55 IF C>5 THEN PRINT'MAKE IT EASY ON YOURSELF--NOT SO MANY.':GOTO 50
 60 FOR O=1 TO C
 65 PRINT'Hit 'return' for test #';O;:INPUT L
 70 A=54302
 80 FOR I=1 TO 30:PRINT:NEXT I
 90 N=0
```

```
100 POKE A,161
110 N=N+1
120 X=INT(4*RND(X)+1)
130 IF N>10 THEN 240
140 ON X GOTO 150,160,170,180
150 I=64:GOTO 190
160 I=1:GOTO 190
170 I=-64:GOTO 190
180 I=-1:GOTO 190
190 A=A+I
200 IF PEEK(A)=161 THEN A=A-I:GOTO 110
210 IF A>55030 OR A<53696 THEN A=A-I:GOTO 110
230 GOTO 90
240 PRINT"IS THIS A:"
245 P=INT(3*RND(1)+1)
250 FOR Q=1 TO 3
255 R=INT(12*RND(1)+1)
257 IF Q=1 THEN M=R
259 IF Q=2 THEN E=R
262 IF M=E THEN 255
265 IF Q=3 THEN F=R:IF F=E OR F=M THEN 255
270 PRINT"(";Q;")";" ";H$(R)
275 F$(Q)=H$(R)
280 NEXT Q
285 INPUT T
287 IF T <1 OR T>3 THEN 285
290 IF T=P THEN PRINT"VERY GOOD!  THAT'S RIGHT!":S=S+1:GOTO 299
295 PRINT"NO, IT'S A ";F$(P):GOSUB 500
299 NEXT O
300 SC=INT((S/C)*100):PRINT:PRINT:PRINT"LET'S SEE NOW."
302 PRINT"YOUR SCORE IS ";SC;"%":PRINT:PRINT"I WOULD
    SAY...":FOR T=1 TO 500:NEX
T T
305 IF SC<=20 THEN PRINT"YOU REALLY ARE A MESS!!":GOTO 390
310 IF SC>20 AND SC<=40 THEN PRINT L$(1):GOTO 390
320 IF SC>40 AND SC<=60 THEN PRINT L$(2):GOTO 390
330 IF SC>60 AND SC<=80 THEN PRINT L$(3):GOTO 390
340 IF SC>80 AND SC<=99 THEN PRINT L$(4):GOTO 390
350 IF SC>99 THEN PRINT L$(5);L$(6):GOTO 390
390 END
400 H$(1)="ROONEY"
401 H$(2)="SICK STAIR CASE"
402 H$(3)="CAT FLYING UPSIDE DOWN"
403 H$(4)="BIRD IN A HALTER TOP"
404 H$(5)="PEANUT BUTTER JET PLANE"
405 H$(6)="DRUNK WITH A FLAT TIRE"
406 H$(7)="WILTED FIRE PLUG"
407 H$(8)="PLATE OF REFRIED BEANS"
408 H$(9)="FROG WITH NUDE"
409 H$(10)="CREEPY CANAL"
410 H$(11)="NAZI ANT HILL"
411 H$(12)="MARTAIN PAGODA"
412 L$(1)="THAT YOUR BRAIN IS A HALF BUBBLE OFF CENTER!"
413 L$(2)="THAT YOU ARE PERFECTLY WELL ADJUSTED AND NORMAL!"
414 L$(3)="THAT YOU SHOULD BE KEPT AWAY FROM SHARP OBJECTS!"
415 L$(4)="THAT TO GET THIS MANY RIGHT YOU ARE GIFTED WITH ESP!"
416 L$(5)="THAT IF YOU GOT ALL OF THEM RIGHT THEN YOU HAVE A VERY "
417 L$(6)="TWISTED  MIND AND SHOULD BE IN A MANAGEMENT POSITION!"
418 RETURN
500 V=INT(4*RND(1)+1)
505 IF V=W OR V=Z THEN 500
510 ON V GOTO 520,600,700,800
520 PRINT"HMMMMMM":FOR T=1 TO 300:NEXT T
530 INPUT"YOU NEVER HAD A PUPPY WHEN YOU WERE A CHILD, DID YOU";W$
540 IF W$="Y" OR W$="YES" THEN PRINT"UH OH!!"
550 IF W$="N" OR W$="NO" THEN PRINT"I THOUGHT SO.  THINGS LIKE THAT SHOW."
560 W=1
570 GOTO 900
600 PRINT"AH, HA!!  THAT IS A VERY REVEALING CHOICE!!!"
610 INPUT"IS YOUR FAVORITE DISH SPINICH AND LIVER WITH BLUE BERRIES";W$
620 IF W$="Y" OR W$="YES" THEN PRINT"YOU HAVE A PROBLEM THAT I CANNOT HELP.
630 IF W$="N" OR W$="NO" THEN PRINT"WHEW, THAT'S VERY GOOD FOR YOU!"
640 Z=2
650 GOTO 900
700 PRINT"YOUR ANSWER IS NOT CORRECT.....BUT IT IS MEANINGFULL."
710 GOTO 900
800 PRINT"YOU'RE CLOSE, BUT NOT RIGHT. YOU SHOULD HAVE COME TO ME SOONER."
810 GOTO 900
900 RETURN
1000 FOR T=1 TO 30:PRINT:NEXT T
1010 PRINT"THE DOCTOR WILL SEE YOU NOW.":FOR T=1 TO 300:NEXT T
1020 FOR T=1 TO 30:PRINT:NEXT T
1030 PRINT"      HELLO.  I'M GOING TO PRESENT A SERIES OF PICTURES"
1040 PRINT"FOR YOU TO LOOK AT.  WHEN I ASK YOU, INDICATE WHICH OF"
1050 PRINT"THE THREE CHOICES IT LOOKS LIKE TO YOU."
1060 PRINT"      BASED ON YOUR ANSWERS, I WILL EVALUATE YOUR MIND."
1070 PRINT"ANSWER THE QUESTIONS AS HONESTLY AS POSSIBLE OR I WILL"
1080 PRINT"CALL YOUR MOMMY ON YOU."
1090 GOTO 50

OK
```



Drawing Pictures Using Printer Plotting

Raymond J. Kalush Jr.
and Alice G. Kalush

Ray and Alice Kalush, 3209 Reo Rd., Lansing, MI 48910.

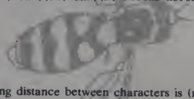
Printer graphics bring to mind wall murals and posters. Most printer pictures require standing back several yards to be fully appreciated. This type of picture usually has a 2:1 distortion of height versus width (i.e. one letter down is about two letters across in distance).

Printer plotting offers an improvement over the standard techniques because pictures can be printed with any horizontal to vertical ratio. One data file can produce many different interpretations of the same picture. The scale can be ½ of the original size or larger. It is also possible to make tall and thin or short and fat pictures.

Since this method provides greater control of where each character is printed, other factors should not louse this up. The main problem is to keep the paper from shifting during printing. The best way to prevent shifting is to only move the paper in one direction and to use a tractor feed. The paper tends to 'slop' when it is moved both up and down. Without the tractor feed the paper tends to 'walk' even unidirectionally. Using friction-feed bidirectionally was disastrous for us.

This technique requires a printer with a 'graphics' mode. It must be possible to adjust the horizontal and vertical increments between characters using software. For example, our Diablo 1620 can be set to any increment of 1/120 inch across or 1/48 inch down. The following control functions are used to set the horizontal and vertical motion for our machine.

```
ESC RS m  CHR$(27)+CHR$(30)+CHR$(M) Define Vertical Motion
ESC US n  CHR$(27)+CHR$(31)+CHR$(N) Define Horizontal Motion
```



The resulting distance between characters is (m-1) * 1/120 inch vertically and (n-1) * 1/48 inch horizontally. Refer to your own printer manual for its control functions.

How Pictures Are Made

The pictures are made by first choosing an original, making a digitization of it and then entering the data into the computer. The first step is to find or make a good line drawing. Next, put a piece of graph paper over it and fill in the squares over the picture lines. When the picture on the graph paper is a good representation of the original then it is ready to enter using a text editor.

The data is entered a line at a time by putting the blank squares and filled squares into the text file as blanks and periods.

The next step is to print the picture using the printer plotting program. The actual height and width parameters will be dependent on your printer. For the Diablo 1620, a small picture can be printed by using 2,4 for height, width and a period for the character. For a larger picture use 3,6 for height, width and an asterisk for the character. Note that the width should be about twice the height for the same horizontal and vertical distance between characters. You can experiment to see what happens with other values.

Some Other Considerations

These pictures have a tendency to take up lots of disk space in their raw form. A method to shrink the size of the data files can be used in conjunction with a modified printing program to save space. One method is to store each line as a list of numbers. This list is formed by taking the number of blanks, the number of dots, the number of blanks, number of dots, ... ending with the number of dots. This method will reduce a 16K file to 3K.

Another method for reducing your data file is to represent it as one decimal digit for each eight binary digits (blanks=0, dots=1). This is less convenient because it is more complicated to produce and then to convert to printable form. This method reduces the file size to about one seventh of the original size.

For example, suppose this was a line of your data file.

1234567890123456789012345678901234567890

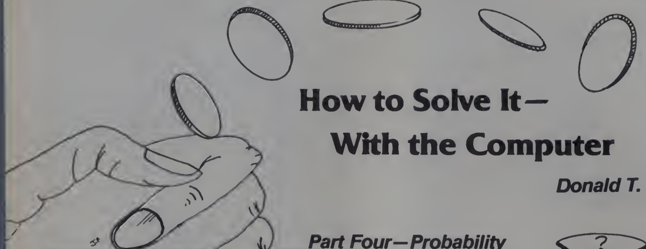
0000iii000iiii0iii0i0i0ii00ii00iiiiii0i0

In the first reduced form the data line would look like:
4,3,3,4,3,1,1,1,1,1,2,2,2,2,6,1,1.

In the first reduced form the data line would look like:
4,3,4,1,3,1,1,1,1,1,2,2,2,2,6,1,1.

In the second reduced form, the data line would look like:
14,61,213,153,250.

In addition to the Diablo 1620 printer, we use a North Star Horizon 2 with double density drives and a Hewlett Packard 2621A terminal. The program was written using the Microsoft Basic 80 Compiler with CP/M. An earlier version of the program was written in CBasic2. □



How to Solve It— With the Computer

Donald T. Piele

Part Four—Probability

> *"To sum up, we can ascertain that, approximately, the frequency of an event is to the number of all observations as the probability of the event is to the probability of the certainty, i.e., to 1. I find this correspondence between facts and logic, between possibility and realization, wonderful, indeed!"*
> *Blaise Pascal (1654)*

The development of the theory of probability is a comparatively young branch of mathematics which historians believe began with a series of letters between Pascal and Fermat in 1654. Motivated by his interest in the gambling problems suggested to him by his friend, the Chevalier de Mere, Pascal formulated many of the fundamental principles of this new science — which we now take for granted. Even the most basic idea of using a number between 0 and 1 to represent the probability of an event had not been formulated before. It was Pascal who suggested "... the most natural procedure is always to assign the number 1 to the complete certainty and to measure the degree of certainty of a random event with the fraction giving the event's share of the complete certainty."[1]

Pascal's letters were published for the first time in the small paperback, *Letters On Probability*,[1] in 1972. At the time they were written, more than 300 years earlier, it was not completely clear whether the study of the randomness was indeed a branch of mathematics. "If by mathematics one understands its traditional capital stock only, geometry, arithmetic, and algebra, there is naturally in this narrow definition no room for any new branch." I agree in this respect with Descartes, however, according to whom every study having for its aim the investigation of measure and order belongs to mathematics irrespective of the object whose measure and order it investigates.

Computers In Mathematics

Today, the network of fields linked to mathematics by this definition are huge. In fact, it is hard to exclude any area of human endeavor that does not ultimately encounter problems of measure and order. In a similar way, computers were originally designed to solve a very narrow class of computational problems in ballistics. Today, less than 35 years later, it is difficult to exclude any area of human endeavor that does not benefit by the rapid computation of measure and order provided by computers.

At the same time that computers have become indispensable tools for solving problems of measure and order, they have expanded the ways we traditionally solve problems and they have provided a completely new environment for developing the art of problem solving.

Donald Piele, University of Wisconsin-Parkside, Kenosha, WI 53141

Programming Activities

This months' activities will be taken from the world of probability. For the beginning student, I will introduce problems that use the random number generator — a function that plays an indispensable role in many computer simulations. The use of relative frequency as a measure of the probability of an event will be explored in problems for the intermediate student. Finally, the average value for the number of tries necessary to write a bug-free program will be explored by the advanced students.

Lesson #4 (Beginning Students)

In the last lesson (#3), I posed the problem of writing a program for the Apple II that would fill the screen with a single color. The intent of this problem was to introduce the FOR-NEXT statement. Now the problem will be to carry out a similar procedure in a completely random fashion.

The commonly used random number function used in Basic is denoted by RND(1). On the Apple II with Applesoft Basic, RDN(1) returns a random number between 0 and 1 every time it is encountered in the program. A simple program illustrates how this works.

```
10 PRINT RND(1)
20 GOTO 10
30 END

RUN
.53345678
.876347891
.293018028
.........
```

Unfortunately, the program must be stopped (Break/CRTL C) to read the numbers because they appear on the screen so fast. For better control of the output, I use the following program which generates 10 random numbers; the numbers multiplied by 10; and then the integer part of the numbers multiplied by 10. The following program illustrates how to use the random number generator to pick single digits at random.

```
10 REM RANDOM NUMBER
20 REM APPLESOFT BASIC
30 FOR I=1 TO 10
40    X=RND(1)
50    PRINT X, 10*X, INT(10*X)
60 NEXT I
70 END

RUN
.34753094       3.4753094      3
.89234103       8.9234103      8
.25345630       2.5345630      2
.........       .........      .
```

Worth A Fortune

Past issues of Creative Computing. What are they worth today? It varies. To a collector, Vol. 1, No. 1 is worth $7 or $8. To a scrap dealer, less than two cents.

But we're not selling old back issues. We're all out.

On the other hand, you know that much of the content of Creative Computing is timeless. The Depth Charge program in Vol. 1, No. 1 is just as challenging today as the day it was written. Walter Koetke's series of five articles on using computers in the classrom are as valid today as the day they first ppeared in print. And scores of people have written about obtaining reprints of Don Piele's classic problem-solving series.

Our Mistake

In our early growth years when we had 5,000 and then 10,000 subscribers we couldn't imagine we would ever need more than 1000 extra copies for back issue sales. That's about what we printed extra. However, by the time we were going into Volume 3, we found our stocks of Volume 1 issues virtually depleted.

Our Solution

So we selected the best material from Volume 1, edited it, put it together in book form and sold it for $8.95, about the same as the six individual issues. Nine months later, we did the same with Volume 2. Then a year and a half later we did it again with Volume 3.

Most other magazines in a high technology field like small computers find their contents are quickly out of date. However, because we've concentrated on applications and software, our content retains its value for a much longer time. Our subscribers know this and retain their copies of Creative Computing long after they've disposed of the more hardware-oriented magazines.

Now you can obtain the best material from the first three years of Creative Computing in book form and the next three years (minus four issues) in the original magazine form.

Our Offer

We have a unique special offer, so pay close attention to this paragraph. (Computer types ought to be able to understand this). If you order any one item below, you pay the full price. If you order any two items, take a 5% discount from the total; any three, take a 10% discount; any four, take a 15% discount, any five, take a 20% discount, and on all six take a whopping 25% discount from the total price.

Best of Creative Computing-Vol 1	$8.95
Best of Creative Computing-Vol 2	8.95
Best of Creative Computing-Vol 3	8.95
Volume 4 (Four issues)	6.00
Volume 5 (Ten issues)	15.00
Volume 6 (Twelve issues)	18.00

Less discount (5% for two items, 10% for three, 15% for four, 20% for five, 25% for all six) Shipping ($2.00 USA, $5.00 foreign)

This isn't really an order form. Cut it out or copy the items you want onto a sheet of paper. Send it with your name and address and payment or credit card number and expiration date (Visa, MasterCard or American Express) to Creative Computing, Morris Plains, NJ 07950. Or call toll-free to 800-631-8112 (in NJ, 201-540-0445).

We guarantee you'll never find a better value in computer applications reading matter. On average you're getting 128 pages of solid information for each $1.00. If you're not completely satisfied after you've read them, send the books or magazines back to us and we'll refund your full purchase price plus the return postage.

creative computing

Morris Plains, NJ 07950
Toll-free **800-631-8112**
(In NJ 201-540-0445)

CIRCLE 300 ON READER SERVICE CARD

for TRS 80 II, Compucolor II, & PET

EVERYTHING YOU NEED TO BRING YOUR BASIC PROGRAMS TO LIFE WITH SOUND!

SOUNDWARE is a complete package

YOU GET A SPEAKER/AMPLIFIER UNIT complete with connectors No wiring or soldering Just plug in!

YOU GET A DEMO PROGRAM with a variety of sample sound effects sirens, laser sounds, tunes!

YOU GET A COMPOSER PROGRAM to help you create your own original sound effects Fun for all ages!

YOU GET AN INSTRUCTION BOOKLET that tell you how to insert sound into your programs 1 year warranty

SOUNDWARE™

	Complete Package	or	Programs & Booklet
ALL PETS & TRS 80 II	$29 95		$14 95
Compucolor II	$39 95		$19 95

SEND FOR FREE CATALOG OF GAMES FOR PET & COMPUCOLOR!!

SEE YOUR DEALER TODAY! Or order direct from CAP by phone or mail VISA & MasterCharge orders include expiration date Add $1 postage & handling per order. $3 for air or COD Arizona residents add tax

CAP Electronics 8462 Hillwood Ln Suite 2 Tucson AZ 85715 (602) 296 4978

CIRCLE 213 ON READER SERVICE CARD

Tarbell Floppy Disc Interface

Designed for Hobbyists and Systems Developers

- Plugs directly into your IMSAI or ALTAIR and handles up to 4 standard single drives in daisy-chain.
- Operates at standard 250K bits per second on normal disc format capacity of 243K bytes.
- Works with modified CP/M Operating System and BASIC-E Compiler.
- Hardware includes 4 extra IC slots, built-in phantom bootstrap and on-board crystal clock. Uses WD 1771 LSI Chip.
- 6-month warranty and extensive documentation.

Tarbell Electronics

950 DOVLEN PLACE, SUITE B
CARSON, CA 90746
(213) 538-4251 • (213) 538-2254

CIRCLE 202 ON READER SERVICE CARD

Solve it, continued...

Integer Basic on the Apple II generates only integers, and thus the RND(X) function has a slightly different meaning. In this case RND(X) generates an integer between 0 and X-1 inclusively every time it is encountered in the program. For example, to perform the same task as above in Integer Basic, use:

```
10 REM RANDOM INTEGER
20 REM INTEGER BASIC
30 FOR I= 1 TO 10
40    PRINT RND(10),
50 NEXT I
60 END
```

As illustrated in lesson #3, positions on the low-resolution graphics screen are located by pairs X,Y where both X and Y are integers between 0 and 39 inclusively. Thus, to pick an integer in this range at random in Applesoft Basic use INT(40*RND(1)). In Integer Basic, RND(40) accomplishes the same thing.

Problem #4 (Beginning Students)

Write a program that fills up the screen with a solid color by plotting the points at random.

Remarks

1. What does it mean to plot points on the screen at random? Discuss this question with the class and see what they think it means. As illustrated above, every point on the screen is represented by a pair of integers (X,Y) where X and Y are between 0 and 39 inclusively. If X and Y are chosen at random in this range then the point X,Y is a random point on the screen.

2. A sample solution in Applesoft Basic is:

```
10 GR
20 COLOR=9
30 X=INT(40*RND(1))
40 Y=INT(40*RND(1))
50 PLOT X,Y
60 GOTO 30
70 END
```

3. For Integer Basic lines 30 and 40 will need to be replaced by

```
30 X=RND(40)

40 Y=RND(40)
```

4. The program is caught in an endless loop which can be terminated by using the familiar CTRL C.

5. After the students have written a Basic solution to this problem I like to pose a number of follow-up questions.

a) What happens if we replace line 30 with "30 X = 20"?
[A vertical line in the middle of the screen is filled at random.]

b) What happens in the original program if we replace line 40 with "40 Y = 20"?
[A horizontal line in the middle of the screen is filled at random.]

c) What happens in the original program if we replace line 50 with "50 PLOT X,X"?
[The diagonal line from the upper left to the lower right of the screen is filled at random.]

d) What happens in the original program if we replace line 60 with "60 GOTO 40"?
[A random horizontal line is filled at random.]

e) What happens in the original program if we replace line 20 with 20 COLOR=INT(16*RND(1)) and line 60 with 60 GOTO 20.
[The screen is plotted at random with random colors.]

Lesson #4 (Intermediate Students)

The simplest example of relative frequency is illustrated by the experiment of flipping a coin a fixed number of times and counting the number of heads and tails that appear. Let H be the variable that counts the number of heads and let N be the total number of tosses. The relative frequency of the occurrence of heads is defined to be H/N. Anyone who has tried this experiment with a fair coin recognizes a certain predictable behavior: The relative frequency H/N is close to 1/2 and seems to get closer the longer the experiment is performed.

This experiment can be simulated on the computer by making the random number generator act like a coin. Random numbers generated in Basic use a procedure that picks out numbers uniformly over the interval (0,1). This means that, on the average, half the numbers are less than .5. Thus, by using the statement

IF RND(1) < .5 THEN H = H + 1

the counter H (heads) is increased by one about 50% of the time. If the chances of a head is P, (0 <= P <= 1), then the statement

IF RND(1) < P THEN H = H + 1

will increment the counter H by one approximately P*100% of the time.

These ideas can be put together into a simple Coin Tossing Experiment. In this experiment the probability of a head is assumed to be .5. A coin is tossed 1000 times and after every 50 tosses the total number of trials (C), the number of heads (H), and the relative frequency (H/C) are reported.

```
100 PRINT "COIN TOSSING EXPERIMENT "
110 PRINT "======================="
120 PRINT "TOSSES",TAB(7),"HEADS",TAB(14),
    "HEADS/TOSSES"
130 FOR I=1 TO 20
140 X=RND(-1)              \REM RANDOM SEED
150 X=RND(0)               \REM RANDOM NUMBER
160 C=C+1                  \REM TOTAL TOSSES
170 IF X < .5 THEN H=H+1
180 IF C/50 <> INT (C/50) THEN 150
190 PRINT C,TAB(7),H,TAB(14),H/C
200 NEXT I
210 END



RUN
COIN TOSSING EXPERIMENT
=======================
TOSSES HEADS  HEADS/TOSSES
 50      30     .6
 100     51     .51
 150     76     .50666667
 200     105    .525
 250     129    .516
 300     152    .50666667
 350     182    .52
 400     203    .5075
 450     227    .50444444
 500     252    .504
 550     274    .49818182
 600     300    .5
 650     328    .50461538
 700     355    .50714286
 750     384    .512
 800     416    .52
 850     438    .51529412
 900     462    .51333333
 950     484    .50947368
 1000    504    .504
```

Remarks

1. The form for the random number generator is not standard in all Basics. This program was written in NorthStar Basic. In Applesoft Basic one should use 150 X = RND(1) to generate the next random number.

2. Line 140 is used to start the program with a random seed value. This is usually handled differently in different Basics. Some Basics use the statement 140 RANDOMIZE. In Applesoft Basic this is equivalent to

140 X = RND(-PEEK(78) 256*PEEK(79)).

3. Line 180 is used to interrupt the experiment after every 50 tosses and print out the current value for C, H, and H/C.

Economic and Ecology Simulations

The Ecology Simulations series are a unique educational tool. They are based on "simulation models" developed by the Huntington Two Computer Project at the State University of New York at Stony Brook under the direction of Dr. Ludwig Braun. The programs and accompanying documentation are written for self-teaching or classroom use and include background material, sample exercises and study guides. Graphic displays were specially developed by Jo Ann Comito at SUNY and Ann Corrigan at Creative Computing. The Ecology Simulations packages are a remarkable educational application of micro-computers.

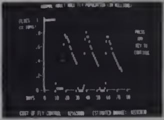
Ecology Simulations-1, CS-3201 (16K)

1. Pop

The POP series of models examines three different methods of population projection, including exponential, S-shaped or logistical, and logistical with low density effects. At the same time the programs introduce the concept of successive refinement of a model, since each POP model adds more details than the previous one.

2. Sterl

STERL allows you to investigate the effectiveness of two different methods of pest control—the use of pesticides and the release of sterile males into the fly population. The concept of a more environmentally sound approach versus traditional chemical methods is introduced. In addition, STERL demonstrates the effectiveness of an integrated approach over either alternative by itself.



3. Tag

TAG simulates the tagging and recovery method that is used by scientists to estimate animal populations. You attempt to estimate the bass population in a warm-water, bass-bluegill farm pond. Tagged fish are released in the pond and samples are recovered at timed intervals. By presenting a detailed simulation of real sampling by "tagging and recovery," TAG helps you to understand this process.

4. Buffalo

BUFFALO simulates the yearly cycle of buffalo population growth and decline, and allows you to investigate the effects of different herd management policies. Simulations such as BUFFALO allow you to explore "What if" questions and experiment with approaches that might be disastrous in real life.

Ordering Information

The series is designed for the 16K TRS-80 Level II and is attractively packaged in a vinyl binder with a complete study guide. *Ecology Simulations-I:* disk CS-3501, cassette 3201. *Ecology Simulations-II:* disk CS-3502, cassette CS-3204. *Social and Economic Simulations:* disk CS-3508, cassette CS-3204. At a modest $24.95 each, the series is an affordable necessity.

To order, send payment plus $1.00 shipping and handling to Creative Computing Software, Dept. ACGG, P.O. Box 789-M, Morristown, NJ 07960. For Faster Service, call in your order toll-free to our order hotline 800-631-8112. In NJ call 201-540-0445.

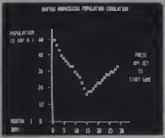
Ecology Simulations-2, CS-3202 (16K)

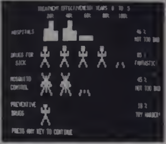
1. Pollute

POLLUTE focuses on one part of the water pollution problem; the accumulation of certain waste materials in waterways and their effect on dissolved oxygen levels in the water. You can use the computer to investigate the effects of different variables such as the body of water, temperature, and the rate of dumping waste material. Various types of primary and secondary waste treatment, as well as the impact of scientific and economic decisions can be examined.

2. Rats

In RATS, you play the role of a Health Department official devising an effective, practical plant to control rats. The plan may combine the use of sanitation and slow kill and quick kill poisons to eliminate a rat population. It is also possible to change the initial population size, growth rate, and whether the simulation will take place in an apartment building or an entire city





3. Malaria

With MALARIA, you are a Health Official trying to control a malaria epidemic while taking into account financial considerations in setting up a program. The budgeted use of field hospitals, drugs for the ill, three types of pesticides, and preventative medication, must be properly combined for an effective control program.

4. Diet

DIET is designed to explore the effect of four basic substances, protein, lipids, calories and carbohydrates, on your diet. You enter a list of the types and amounts of food eaten in a typical day, as well as your age, weight, sex, health and a physical activity factor. DIET is particularly valuable in indicating how a diet can be changed to raise or lower body weights and provide proper nutrition.

Social and Economic Simulations CS-3204 (16K)

1. Limits

LIMITS is a micro-computer version of the well known "Limits to Growth" project done at MIT. It contains a model of the world that is built of five subsystems (population, pollution, food supply, industrial output, and resource usage) linked together by six variables: birth rate, death rate, pollution generation, resource usage rate, industrial output growth rate, and food production rate.

2. Market

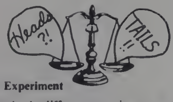
Market allows two or more people to play the roles of companies who are competing for the market for a particular product: in this case, bicycles.

Each player makes marketing decisions quarterly including the production level, the advertising budget, and the unit price of the product for his/her company.

3. USPop

USPOP allows the user to study many aspects of the United States' human demography (population change) including population growth, age and sex distribution. USPOP makes population projections and investigates the consequences of many different demographic changes.

Solve it, continued...



Geometric Experiment

A completely different experiment can be performed by observing a series of coin tosses. Suppose instead of counting the number of heads and tails we count the number of tosses until a head occurs for the first time. This defines a new experiment called the First Head Experiment. Every time the experiment is performed we observe a number — which represents the total number of tosses needed to get a head for the first time. Thus, the outcomes of this experiment are recorded as follows.

EXPERIMENT	OUTCOME
H	1
TH	2
TTH	3
. . .	

In this experiment, if we assume that the probability of getting a head on each toss is P, then the probability for a tail is 1-P. Under these conditions, this experiment has a Geometric distribution which I will elaborate later.

Problem #4 (Intermediate Student)
First Head Experiment

Write a program that will simulate the experiment of tossing a coin until the first head occurs. The program must allow for the entry of the probability P of a head on each toss. Perform the experiment 1000 times and keep track of the number of experiments that end after T tosses, T = 1,2,3... M. (M is the largest number of tosses needed so far.) Print out a table after every 100 experiments showing the distribution of the outcomes that end after T attempts and the relative frequency for each value of T.

Remarks:

1. The heart of this simulation is a routine that will simulate the tossing of the coin until the first head. Suppose the counter T is used to store the number of trials until the first head and that P is the probability of a head on each toss. The following few lines simulate this experiment.

```
......
240 T = 0                 T = TRAIL #
250 X = RND(0)            TOSS COIN
260 T = T + 1             INCREMENT TRIAL
270 IF X > P THEN 250     TOSS IS A TAIL
....                      1st HEAD OCCURS
```

2. The 250-270 loop terminates whenever the random number RND(0) is less than P. This corresponds to the occurrence of a head.

3. It is convenient to use an array, such as C(T), to keep track of the number of times the experiment ends after T trials. Each time the experiment ends after T tosses, it is counted by

290 C(T) = C(T) + 1.

4. If N is equal to the total number of experiments performed, then C(T)/N is equal to the relative frequency of the number of experiments that end after T trials.

5. If N/100 = INT(N/100), then N is a multiple of 100. A version of this test is used to print out the distribution after every 100 experiments.

6. A sample program that solves this problem is as follows.

The Sinclair ZX80 is innovative and powerful. Now there's a magazine to help you get the most out of it.

Get in sync

SYNC magazine is different from other personal computing magazines. Not just different because it is about a unique computer, the Sinclair ZX80 (and kit version, the MicroAce). But different because of the creative and innovative philosophy of the editors.

A Fascinating Computer

The ZX80 doesn't have memory mapped video. Thus the screen goes blank when a key is pressed. To some reviewers this is a disadvantage. To our editors this is a challenge. One suggested that games could be written to take advantage of the screen blanking. For example, how about a game where characters and graphic symbols move around the screen while it is blanked? The object would be to crack the secret code governing the movements. Voila! A new game like Mastermind or Black Box uniquely for the ZX80.

We made some interesting discoveries soon after setting up the machine. For instance, the CHR$ function is not limited to a value between 0 and 255, but cycles repeatedly through the code. CHR$ (9) and CHR$ (265) will produce identical values. In other words, CHR$ operates in a MOD 256 fashion. We found that the "=" sign can be used several times on a single line, allowing the logical evaluation of variables. In the Sinclair, LET X=Y=Z=W is a valid expression.

Or consider the TL$ function which strips a string of its initial character. At first, we wondered what practical value it had. Then someone suggested it would be perfect for removing the dollar sign from numerical inputs.

Breakthroughs? Hardly. But indicative of the hints and kinds you'll find in every issue of **SYNC**. We intend to take the Sinclair to its limits and then push beyond, finding new tricks and tips, new applications, new ways to do what couldn't be done before. **SYNC** functions on many levels, with tutorials for the beginner and concepts that will keep the pros coming back for more. We'll show you how to duplicate commands available in other Basics. And, perhaps, how to do things that can't be done on other machines.

Many computer applications require that data be sorted. But did you realize there are over ten fundamentally different sorting algorithms? Many people settle for a simple bubble sort perhaps because it's described in so many programming manuals or because they've seen it in another program. However, sort routines such as heapsort or Shell-Metzner are over 100 times as fast as a bubble sort and may actually use less memory. Sure, 1K of memory isn't a lot to work with, but it can be stretched much further by using innovative, clever coding. You'll find this type of help in **SYNC**.

Lots of Games and Applications

Applications and software are the meat of **SYNC**. We recognize that along with useful, pragmatic applications, like financial analysis and graphing, you'll want games that are fun and challenging. In the charter issue of **SYNC** you'll find several games. Acey Ducey is a card game in which the dealer (the computer) deals two cards face up. You then have an option to bet depending upon whether you feel the next card dealt will have a value between the first two.

In Hurkle, another game in the charter issue, you have to find a happy little Hurkle who is hiding on a 10 X 10 grid. In response to your guesses, the Hurkle sends our a clue telling you in which direction to look next

One of the most ancient forms of arithmetical puzzle is called a "boomerang." The oldest recorded example is that set down by Nicomachus in his *Arithmetica* around 100 A.D. You'll find a computer version of this puzzle in **SYNC**.

Hard-Hitting, Objective Evaluations

By selecting the ZX80 or MicroAce as your personal computer you've shown that you are an astute buyer looking for good performance, an innovative design and economical price. However, selecting software will not be easy. That's where **SYNC** comes in. **SYNC** evaluates software packages and other peripherals and doesn't just publish manufacturer descriptions. We put each package through its paces and give you an in-depth, objective report of its strengths and weaknesses.

SYNC is a Creative Computing publication. **Creative Computing** is the number 1 magazine of software and applications with nearly 100,000 circulation. The two most popular computer games books in the world, *Basic Computer Games* and *More Basic Computer Games* (combined sales over 500,000) are published by Creative Computing. Creative Computing Software manufactures over 150 software packages for six different personal computers.

Creative Computing, founded in 1974 by David Ahl, is a well-established firm committed to the future of personal computing. We expect the Sinclair ZX80 to be a highly successful computer and correspondingly, **SYNC** to be a respected and successful magazine.

Order SYNC Today

Right now we need all the help we can get. First of all, we'd like you to subscribe to **SYNC.** Subscriptions cost $10 for one year (6 issues), $18 for two years (12 issues) or, if you really want to beat inflation, $24 for three years (18 issues). Send to the address below or call our toll-free number, **800-631-8112** (in NJ 201-540-0445) to put your subscription on your MasterCard, Visa or American Express card. Canadian and other foreign surface subscriptions are $15 per year or $27 for two years. We guarantee your satisfaction or we will refund the unfulfilled portion of your subscription.

Needless to say, we can't fill up all the pages without your help. So send in your programs, articles, hints and tips. Remember, illustrations and screen photos make a piece much more interesting. Send in your reviews of peripherals and software too—but be warned: reviews must be in-depth and objective. We want you to respect what you read on the pages of **SYNC** so be honest and forthright in the material you send us. Of course we pay for contributions—just don't expect to retire on it.

The exploration has begun. Join us.

The magazine for Sinclair ZX80 users

SYNC

39 East Hanover Avenue
Morris Plains, NJ 07950, USA

Toll free **800-631-8112**
(In NJ 201-540-0445)

CIRCLE 215 ON READER SERVICE CARD

```
100 PRINT "FIRST HEAD EXPERIMENT"
110 PRINT "====================="
120 PRINT "THE TOSS OF A COIN RESULTS IN EITHER A HEAD OR A TAIL."
130 PRINT "THE PROBABILITY OF HEADS IS THE NUMBER P."
140 PRINT "AN EXPERIMENT CONSISTS OF TOSSING A COIN UNTIL THE FIRST HEAD OCCURS."
150 PRINT "THE OUTCOME OF THE EXPERIMENT IS THE NUMBER OF TOSSES NEEDED."
160 PRINT
170 PRINT "THIS EXPERIMENT IS PERFORMED 1000 TIMES AND THE DISTRIBUTION "
180 PRINT "OF TOSSES IS REPORTED AFTER EVERY 100 EXPERIMENTS."
190 PRINT
200 INPUT "ENTER P THE PROBABILITY OF A HEAD ON EACH TOSS, P = ",P
210 DIM C(100)
220 X=RND(-1)                  \ REM RANDOM SEED
230 FOR J=1 TO 10
240 T=0                        \ REM T = # OF TRIALS UNTIL FIRST HEAD
250 X=RND(0)
260 T=T+1
270 IF X>P THEN 250            \ REM THE TOSS IS A TAIL
280 N=N+1                      \ REM EXPERIMENT COUNTER
290 C(T)=C(T) + 1              \ REM OUTCOME COUNTER
300 IF T>M THEN M=T            \ REM M = MAX # OF TRIALS IN ANY EXPERIMENT
310 IF N/100<>INT(N/100) THEN 240
320                           REM  PRINT OUT OF RESULTS
330 PRINT
340 PRINT "TRIAL",TAB(10),"# OF EVENTS",TAB(25),"RELATIVE FREQUENCY"
350 FOR I=1 TO M
360 PRINT I,TAB(10),C(I),TAB(25),C(I)/N
370 NEXT I
380 PRINT "=========================================="
390 PRINT "TOTALS",TAB(10),N,TAB(25),1
400 NEXT J
410 END

RUN
FIRST HEAD EXPERIMENT
=====================
THE TOSS OF A COIN RESULTS IN EITHER A HEAD OR A TAIL.
THE PROBABILITY OF HEADS IS THE NUMBER P.
EACH EXPERIMENT CONSISTS OF TOSSING A COIN UNTIL THE FIRST HEAD OCCURS.
THE OUTCOME OF THE EXPERIMENT IS THE NUMBER OF TOSSES NEEDED.

THIS EXPERIMENT IS PERFORMED 1000 TIMES AND THE DISTRIBUTION
OF TOSSES IS REPORTED AFTER EVERY 100 EXPERIMENTS.

ENTER P THE PROBABILITY OF A HEAD ON EACH TOSS, P =  .5

(ONLY THE LAST DISTRIBUTION IS PRINTED HERE)
TRIAL      # OF EVENTS    RELATIVE FREQUENCY
 1          501            .501
 2          245            .245
 3          127            .127
 4          62             .062
 5          33             .033
 6          17             .017
 7          3              .003
 8          6              .006
 9          5              .005
 10         1              .001
==========================================
TOTALS      1000           1
```

Lesson #4 (Advanced Students)

The First Head Experiment serves as an introduction to a slightly different experiment which I will call the Bug-Free Program Experiment. As everyone who has ever tried to write a computer program knows, the chances that it will run correctly on the first trial is definitely below 1. We also know from experience that the chances of eliminating bugs usually improves the more times the program is modified and tried again. I will assume, for the purposes of this discussion, that the chances that a program will run perfectly (bug-free) on the Tth trail is equal to T/(T+1). Thus,

```
P(S) = T/(T+1) and

P(F) = 1/(T+1),
```

where P(S) stands for the probability of a successful program on the Tth trial and P(F) represents the probability of a failure on the Tth trial.

This experiment is similar to the First Head Experiment described above, with one big difference. Now the probability of a success depends on the number of times the experiment has been *tried*. The outcome of this experiment is still the number of trials until the first success.

With this model it seems natural to wonder: How many trials, on the average, will it take to produce a bug-free program? In the language of probability, what is the expected value of the outcomes in this experiment?

The average value of the outcomes of an experiment is easy to compute. For example, assume that the outcomes of an experiment are the integers T, T = 1,2. . . M. Also, assume that C(T) counts the number of times T occurred. Then the average number of times that T occurs in N trials is (C(1)+2*C(2)+. . . M*C(M))/N.

Problem #4 (Advanced Students)

Bug-Free Program Experiment

Write a program that simulates the process of writing a program until it runs bug-free. The probability of success on the Tth trial should equal T/(T+1). Perform the experiment 1000 times and keep track of the number of programs that run after T trials where T = 1,2,3. . .M. (M is the largest number of attempts needed so far.) Print out a table, after every 100 experiments, showing the distribution of outcomes that end after T attempts, the relative frequency for each value of T, and the average value of T.

Remarks:

1. It would be advisable for students to write the First Head Experiment program first. With a few changes and modifications this program can be used to solve the Bug-Free Program Experiment. For example: Line 200 will no longer be needed, since the probability of success on the Tth trial is no longer fixed at P. Line 270 should be changed to read: IF X > T/(T=1) THEN 250.

CIRCLE 192 ON READER SERVICE CARD

CIRCLE 185 ON READER SERVICE CARD

CIRCLE 190 ON READER SERVICE CARD

CIRCLE 193 ON READER SERVICE CARD

2. To compute the average outcome in N experiments, the student will need to total all the outcomes. This can be done with a statement such as S = S + I*C(I) placed in the printout loop. The average value then is S/N.

3. This same average number computation could also be added into the First Head Experiment.

4. For added interest, ask students to keep track of how many attempts were made to write the Bug-Free Experiment program. Have the class make a chart of this distribution and compare it with the outcome of the Bug-Free Program experiment.

Postscript

This section is added for those interested in a more detailed mathematical treatment of the two experiments.

The First Head Experiment is an application of the Geometric distribution.[2] Before I go further, some notation may be helpful.

Notation	Meaning
H	Heads
T	Tails
P(H)= p	Probability of heads = p
P(T)= 1-p	Probability of tails = 1-p
q	q = 1-p
TTTTH	First head is on 5th trial.
P(TTTTH)	Probability of the event.
X=5	First head is on 5th trial.
P(X=5)	Probability of the event.
E(X)	Average value of X.


```
100 PRINT "BUG-FREE PROGRAM EXPERIMENT"
110 PRINT "==========================="
120 PRINT "THE PROBABILITY THAT A PROGRAM IS BUG-FREE DEPENDS ON "
130 PRINT "THE NUMBER OF TIMES IT HAS BEEN REWRITTEN."
140 PRINT "EACH TIME IT IS RUN THE PROBABILITY THAT IT IS BUG "
150 PRINT "FREE IS EQUAL TO  T/(T+1) WHERE T IS THE TRIAL NUMBER."
160 PRINT
170 PRINT "AN EXPERIMENT CONSISTS OF EDITING THE PROGRAM UNTIL IT RUNS."
180 PRINT "THE OUTCOME IS THE NUMBER OF TRIALS NEEDED TO MAKE IT WORK."
190 PRINT "THIS EXPERIMENT IS PERFORMED 1000 TIMES AND THE DISTRIBUTION "
200 PRINT "OF OUTCOMES IS REPORTED AFTER EVERY 100 EXPERIMENTS."
210 PRINT "THE AVERAGE NUMBER OF TRIALS NEEDED TO PRODUCE A BUG-FREE PROGRAM"
220 PRINT "IS COMPUTED."
230 PRINT
240 DIM C(100)
250 X=RND(-1)
260 FOR J=1 TO 10
270 T=0                        \ REM T = # OF TRIALS UNTIL PROGRAM RUNS
280 X=RND(0)
290 T=T+1
300 IF X>T/(T+1) THEN 280      \ REM THE PROGRAM HAS A BUG
310 N=N+1                      \ REM EXPERIMENT COUNTER
320 C(T)=C(T) + 1              \ REM OUTCOME COUNTER
330 IF T>M THEN M=T            \ REM M = MAX # OF TRIALS IN ANY EXPERIMENT
340 IF N/100<>INT(N/100) THEN 270
350                       REM  PRINT OUT OF RESULTS
360 PRINT
370 PRINT "TRIAL",TAB(10),"# OF EVENTS",TAB(25),"RELATIVE FREQUENCY"
380 S=0
390 FOR I=1 TO M
400 PRINT I,TAB(10),C(I),TAB(25),C(I)/N
410 S=S+J*C(I)                 \ REM S = SUM OF ALL OUTCOMES
420 NEXT I
430 PRINT "========================================="
440 PRINT "TOTALS",TAB(10),N,TAB(25),1
450 PRINT "AVERAGE VALUE ",S/N
460 NEXT J
470 END


RUN
BUG-FREE PROGRAM EXPERIMENT
===========================
THE PROBABILITY THAT A PROGRAM IS BUG-FREE DEPENDS ON
THE NUMBER OF TIMES IT HAS BEEN REWRITTEN.
EACH TIME IT IS RUN THE PROBABILITY THAT IT IS BUG
FREE IS EQUAL TO  T/(T+1) WHERE T IS THE TRIAL NUMBER.

AN EXPERIMENT CONSISTS OF EDITING THE PROGRAM UNTIL IT RUNS.
THE OUTCOME IS THE NUMBER OF TRIALS NEEDED TO MAKE IT WORK.
THIS EXPERIMENT IS PERFORMED 1000 TIMES AND THE DISTRIBUTION
OF OUTCOMES IS REPORTED AFTER EVERY 100 EXPERIMENTS.
THE AVERAGE NUMBER OF TRIALS NEEDED TO PRODUCE A BUG-FREE PROGRAM
IS COMPUTED.

(ONLY THE LAST DISTRIBUTION IS PRINTED HERE.)
TRIAL     # OF EVENTS    RELATIVE FREQUENCY
 1         500            .5
 2         335            .335
 3         118            .118
 4         39             .039
 5         7              .007
 6         1              .001
=========================================
TOTALS     1000           1
AVERAGE VALUE  1.721
```

Since all tosses of the coin are independent of one another it is true that

$$P(TTTTH) = P(T)P(T)P(T)P(T)P(H) = q^4 p.$$

If X stands for the number of tosses until the first head appears, the following probability table applies.

Outcome X	Probability
1	p
2	$q^1 p$
3	$q^2 p$
....	...
i	$q^{i-1} p$

The average or expected outcome is determined by weighing each outcome by its probability.

$$E(X) = p + 2pq + 3pq^2 + \ldots\ldots ipq^{i-1} + \ldots$$

It can be shown, [2], that the sum of this series is 1/p. Thus in the First Head Experiment with p=1/2, the average outcome E(X) = 2.

Bug-Free Program Experiment

The computation of the probability of each outcome is different for the Bug-Free Program Experiment. Let F represent failure and S represent success on each trial. The corresponding probabilities, P(F) and P(S), depend on the number of trails T performed

$$P(S) = T/(T+1), \quad P(F) = 1/(T+1).$$

Thus the computation of the probability for the first success occurring on the 5th trial is

$$= \frac{1}{2} \cdot \frac{1}{3} \cdot \frac{1}{4} \cdot \frac{1}{5} \cdot \frac{5}{6} = \frac{5}{6!} \text{ (!=factorial)}$$

If X stands for the number of trials of the program until the first success, then the following probability table applies.

Outcome X	Probability
1	1/2
2	1/3
3	1/8
.	...
i	i/(i+1)!

The average or expected number of trials for this experiment is computed by weighing each outcome with its probability.

$$E(X) = 1/2 + 2(1/3) + 3(1/8) + \ldots + i(i/(i+1)!) + \ldots$$

Each term of this series may be rewritten using the identity

$$i(i/(i+1)!) = i/i! - i/(i+1)!$$

After combining terms, the series can be reduced to

$$E(X) = 1 + 1/2! + 1/3! + 1/4! + \ldots + 1/i! + \ldots$$

This familiar series is equal to e-1, where e is approximately 2.7182817. Thus, using this model, the average number of times that a computer program will need to be debugged is 1.7182817.

This is probably a well known exercise in probability, but for me it came as a delightful surprise. □

REFERENCES

1. Rényi, Alfréd. *Letters On Probability*. Wayne State University Press. Detroit, 1972.
2. Meyer, Paul. *Introductory Probability And Statistical Applications*. p. 171. Addison-Wesley Publishing Co., 1970.

CALIFORNIA

Rainbow Computing, Inc.—9719 Reseda Blvd., Northridge 91324; (213) 349-5560. 10-7 Tues-Fri, 10-5 Sat, 12-5 Sun. Apple, DEC, and Atari. Authorized Sales and Service.

D.E.S. Data Equipment Supply—8315 Firestone, Downey 90241. (213) 923-9361. 7 days. Commodore PET specialists. Hardware Software, Books, Mags, Supplies, In House Maintenance.

Advance Data Concepts—2280 Diamond Blvd., Concord, 94520; (415) 671-9016. 9-5 Mon-Fri. Vector-Graphic, CP/M Software Headquarters-User's Group.

CONNECTICUT

Computerworks—1439 Post Rd., East Westport 06880; (203) 255-9096. 12-6 Tues-Fri, 12-9 Thu, 10-5 Sat.

FLORIDA

AMF Electronics—11158 N. 30th St, Tampa 33612; (813) 971-4072. 10-6 Mon-Sat. Apple Computer Sales & Service; TRS-80, Apple Software & Peripherals; S-100 boards, computer parts & books.

GEORGIA

Atlanta Computer Mart—5091 Buford Hwy, Atlanta 30340; (404) 455-0647. 10-6 Mon-Sat

ILLINOIS

Computer Junction—543 S. York Rd., Elmhurst 60126; (312) 530-1125. Mon & Thu 9:30-8:30 pm; Tues-Sat 9:30-5:30; Sun 12-4:30.

To include your store in Creative Computing's Retail Roster, call the Advertising Department at (201) 540-9168.

ComputerLand/Downers Grove—136 Ogden Ave, Downers Plaza 60515; (312) 964-7762. 10-6 Mon-Sat, 10-8 Tue, Thu.

Data Domain of Schaumburg—1612 E. Algonquin Rd., Schaumburg 60195; (312) 397-8708. 12-9 Tues-Fri, 11-5 Sat. Apple, Alpha Micro, Hewlett-Packard Calculators. Largest book and magazine selection.

Lillipute Computer Mart, Inc.—4446 Oakton, Skokie 60076; (312) 674-1383. M-F 10:30-8 pm, Sat 10-6. We sell Cromemco, Gimix, Bell & Howell, NorthStar and others. Starting our fifth year in business.

Video Etc.—465 Lake Cook Plaza, Deerfield 60015; (312) 498-9669; Open Everyday. Strong software support for Apple, Atari.

KENTUCKY

ComputerLand of Louisville—10414 Shelbyville Rd, Louisville 40223; (502) 245-8288. 10-5:30.

MASSACHUSETTS

Computer Mart, Inc.—1395 Main St, Waltham 02154; (617) 899-4540. 11-6 Tue-Sat. Atari, Heath, NEC, SWTP & S-100 bus systems; Word Processing Specialists.

Neeco—679 Highland Ave, Needham 02194; (617) 449-1760. 9-5:30 Mon-Fri. Commodore, Apple, Superbrain, TI 99/4.

MICHIGAN

The Computer Center—31 East 31st St. New York 10016; (212) 889-8130. 10-7 Mon-Fri, 11-6 Sat, 10-8 Thurs.

NEVADA

Home Computers—1775 E. Tropicana #6, Las Vegas 89109 (702) 798-1022. 10-7 Mon-Sat. Apple, Commodore, Atari, AIM 65, (Books) Sales & Service.

NEW JERSEY

Computernook—Rt. 46, Pine Brook Plaza, Pine Brook 07058; (201) 575-9468. 10-6:30 MTWS, 10-8 Thurs., Fri. Apple/Commodore Authorized dealer.

Stonehenge Computer Shop—89 Summit Avenue, Summit 07901; (201) 277-1020. 10 am - 6:30 pm Mon-Sat. Apple/Bell & Howell/Commodore Authorized Dealer, Sales and Service.

NEW YORK

Computer Center—28251 Ford Rd. Garden City (313) 425-2470 Books, Magazines, Hardware and Software for Apple, Northstar, TRS80, & Pet.

OHIO

Abacus II—1417 Bernath Pkwy., Toledo 43615; (419) 865-1009.

The Basic Computer Shop—2671 W. Market St, Akron 44313; (216) 867-0808. 10-6 Mon-Sat.

Micro Mini Computer World—74 Robinwood Ave., Columbus 43213; (614) 235-5813/6058. 11-7 Tues-Sat. Authorized Apple/Commodore dealer. Sales, Service, Business Software.

PENNSYLVANIA

Personal Computer Corp.—24-26 W. Lancaster Ave, Paoli 19301; (215)647-8463. 10-6 Mon-Fri, 10-8 Wed, 10-5 Sat.

Special editions for Apple, Atari and TRS-80 Computers.

Programming is Child's Play

Hey kids, are the folks out of the room? Good, 'cause I've got a secret to tell you. You know that computer they fuss over? Well, kid, between you and me, this whole programming thing is a lot simpler than they realize.

What's that? Sure, you can learn. Just get a copy of **Computers For Kids.** It's a super book, and it tells you everything you need to know. Huh? You have an Apple? No problem. There's a version just for the Apple. One for the TRS-80 and one for the Atari too, with complete instructions for operating and programming.

The book will take you through everything programmers learn. Its easy to understand and the large type makes it easy to read. You'll find out how to put together a flowchart, and how to get your computer to do what you want it to do. There's a lot to learn, but **Computers For Kids** has 12 chapters full of information. You'll even learn how to write your own games and draw pictures that move.

Just so the folks and your teachers won't feel left out, there's a special section for them. It gives detailed lesson ideas and tells them how to fix a lot of the small problems that might pop up. Hey, this book is just right for you. But you don't have to take my word on that. Just listen to what these top educators have to say about it:

Donald T. Piele, Professor of Mathematics at the University of Wisconsin-Parkside says, "**Computers For Kids** is the best material available for introducing students to their new computer. It is a perfect tool for teachers who are learning about computers and programming *with* their students. Highly recommended."

Robert Taylor, Director of the Program in Computing and Education at Teachers College, Columbia University states, "it's a good idea to have a book *for chidren*."

Not bad, huh? Okay, you can let the adults back in the room. Don't forget to tell them **Computers For Kids** by Sally Greenwood Larsen cost only \$3.95. And tell them you might share it with them, if they're good. Specify edition on your order: TRS-80 (12H); Apple (12G); Atari (12J).

Your local computer shop should carry **Computers For Kids**. If they don't ask them to get it or order by mail. Send \$3.95 payment plus \$1.00 shipping and handling to Creative Computing Press, P.O. Box 789-M, Morristown, NJ 07960.

creative computing press

CIRCLE 350 ON READER SERVICE CARD

SUPERBRAIN

32K or 64K (Double or Quad Density units available). Uses two Z-80 CPU's. Commercial-type terminal with 12" monitor. Dual double density minifloppies. Over 350 kilobytes of storage (twice that with quad density drives). Two serial RS232 ports, I/O ports standard. Expandable with optional S-100 S-100 interface. Comes with CP/M™ 2.2 operating system. MiniMicroMart includes BASIC interpreter and can supply a wide range of CP/M Development and Application software.

w/32K Double Density, List \$2995	**\$2685**
w/64K Double Density, List \$3345	**\$2883**
w/64K Quad Density, List \$3995	**\$3595**
64K Special Quad Version	**\$3395**

INTERSYSTEMS
formerly ITHACA AUDIO

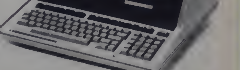
DPS-1, List \$1795

Call for Price!

The new Series II CPU Board features a 4 MHz Z-80A CPU and a full-feature front panel. 20-slot actively terminated motherboard, with 25 amp power supply (50/60 Hz operation, incl. 68 cfm fan).

COMPLETE SYSTEM with InterSystem 64K RAM, I/O Board w/priority interrupt and double density disk controller board. Full 1-year warranty, List \$3595

HEWLETT-PACKARD HP-85A



F.O.B. shipping point. All prices subject to change and all offers subject to withdrawal without notice. Advertised prices are for prepaid orders. Credit card and C.O.D. 2% higher. C.O.D. may require deposit.

— WRITE FOR FREE CATALOG —

MiniMicroMart

1618 James Street
Syracuse, NY 13203 (315) 422-4467

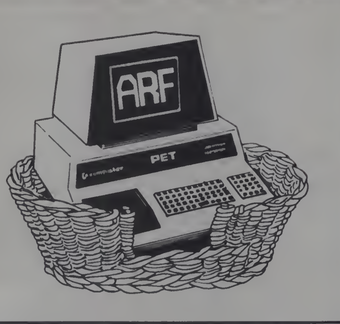
CIRCLE 157 ON READER SERVICE CARD

EENY
MEENY
MINEY
©Creative Computing
miller

"'Que Sera Sera' isn't good enough!"

Personal Electronic Transactions

by Gregory Yob



I am happy to hear from you, and encourage your correspondence. I will try to acknowledge all correspondence, and a SASE makes things easier for both of us. Please send your letters to "Personal Electronic Transactions" c/o PO Box 354, Palo Alto, CA 94301.

A List Program For Non-PET Printers

A long time ago, as time is seen in small computers, I took a look at how the Basic was stored in the PET, and came up with a program which listed itself. (See Sept 79 Column) In the last several columns an undesirable number of typos have crept in, probably because I am too lazy to re-enter a program to see if it works. To combat this, I will now use genuine listings (via my listing program) for any programs longer than a few lines.

Program 1 shows my listing program as set to list itself. This program should work with almost any full ASCII printer, and with minor changes, on any printer that can print the letters and numbers, such as a Teletype (TM). Operation of the program is simple. First, LOAD the program you want to list. Then, using the Programmer's Toolkit APPEND (or Disk-O-Pro's MERGE), append the PRINTER LIST program. When your printer is ready, a RUN 63000 will do the job. The screen will clear and ask for:

PGM NAME:?

which lets you enter a little line of some kind. The printer will then list the program, NOT including the Printer List portion.

PRINTER LIST is written to save as much space as possible (Some of you out there will find ways to save a few bytes, I am sure.) and as a result will look a bit obscure. Let's take a short walk through the program to shed some light on this. If you don't know how Basic is put together, I refer you to the Sept 79 column mentioned earlier.

FNF in Line 63000 computes the value of a pointer at X. That is, if addresses X and X+1 are a pointer, FNF figures out the pointer's value. P is set up to the first pointer in a Basic program which is usually at 1025. (Some creative types will change the Bottom-of-Basic pointer to pick up the 191 bytes in the Second Cassette Buffer. Be duly warned.) T$ is set to the tilde (~) character in ASCII.

The unusual string in the DATA statement,\← ← \, was chosen to be a string which is not likely to appear in any program. Line 63010 reads any DATA lines in the program to be listed and will stop with the DATA pointer just past this unusual string. This is a cheap way to do the equivalent of RESTORE 63010 which is not available on the PET.

The string P$ is set to the ASCII Lower Case letters "sp", Q$ is set to "pi". There is no direct way to set these characters in the PET, as the values 96 to 127 are not enterable via the keyboard. In Line 63020, the string array S$ is defined and loaded with the 75 Basic token names END ... GO. These names are in the DATA statements in Lines 63050-63090. (Note: The "new" PET ROMs have the GO token

```
PROGRAM 1 - PRINTER LIST
 63000  DEFFNF(X)=PEEK(X)+256*PEEK(X+1):P=1025:T$=CHR$(126):DATA\__\
 63010  READN$:IFN$<>"\__\"THEN63010
 63020  P$=CHR$(115)+CHR$(112):DIMS$(75),C%(9),C$(9):FORJ=0TO75:READS$(J):NEXT
 63030  Q$=CHR$(112)+CHR$(105):R$=CHR$(13):S$="sp ":FORJ=0TO9:READC%(J),N$
 63040  FORK=1TOLEN(N$):C$(J)=C$(J)+CHR$(ASC(MID$(N$,K))+32):NEXTK,J
 63050  DATAEND,FOR,NEXT,DATA,INPUT#,INPUT,DIM,READ,LET,GOTO,RUN,IF,RESTORE,GOSU
B
 63060  DATARETURN,REM,STOP,ON,WAIT,LOAD,SAVE,VERIFY,DEF,POKE,PRINT#,PRINT,CONT
 63070  DATALIST,CLR,CMD,SYS,OPEN,CLOSE,GET,NEW,TAB(,TO,FN,SPC(,THEN,NOT,STEP,+,
-
 63080  DATA*,/,↑,AND,OR,>,=,<,SGN,INT,ABS,USR,FRE,POS,SQR,RND,LOG,EXP,COS,SIN,T
AN
 63090  DATAATN,PEEK,LEN,STR$,VAL,ASC,CHR$,LEFT$,RIGHT$,MID$,GO
 63100  DATA147,CLR,19,HM,145,UP,17,DN,29,RT,157,LFT,148,INS,20,DEL,18,RVS,146,O
FF
 63110  INPUT"clr PGMsp NAME:";N$:OPEN 4,4:PRINT#4,CHR$(30)R$R$R$R$N$R$
 63120  N=FNF(P+2):T=P+4:P=FNF(P):IFN=63200THENEND
 63130  PRINT#4,N"sp ";:Q=0:GOSUB63140:PRINT#4:GOTO63120
 63140  C=PEEK(T):IFC=0THENRETURN
 63150  GOSUB63160:GOTO63140
 63160  T=T+1:IFC=34THENQ=Q+1AND1
 63170  IFQTHEN63220
 63180  IFC=255THEN63230
 63190  IFC>203THENC=42
 63200  IFC<128THENPRINT#4,CHR$(C);:RETURN
 63210  PRINT#4,S$(C-128);:RETURN
 63220  IFC=32THENPRINT#4,P$S$;:RETURN
 63230  IFC=255THENPRINT#4,Q$S$;:RETURN
 63240  IFC>31ANDC<128THENPRINT#4,CHR$(C);:RETURN
 63250  IFC>159THENPRINT#4,T$;CHR$(C);:RETURN
 63260  FORJ=0TO9:IFC=C%(J)THENPRINT#4,C$(J)S$;:RETURN
 63270  NEXT:RETURN
```

NOTE THAT LINES 63050, 63070, 63080 AND 63100 WILL FIT ON YOUR PET, PROVIDED THAT YOU DON'T ENTER ANY SPACES.

APPLE DISKS

Each Containing Multiple Programs
TRS Programs Available Early '81

BASIC TUTOR SERIES

This series of disks is designed for the novice programmer and for those interested in learning how to program the **APPLE** Computer in **Basic.**

Basic Tutor I — Introduction to programming commands
Basic Tutor II — Continuation of Tutor I with the introduction of **Graphic Commands** in low and high resolution graphics modes
Basic Tutor III — Introduction to the study of **Text Files** and the basis for most of the "data processing" of strings and words
Basic Tutor IV — Studies in some simple methods of creating computer sounds and **music** composition
Basic Tutor V — This disk includes some of the program studies that have been created by "novice" programmers, and are intended for your use and study
Basic Tutor VI — Studies in the intermediate level of programming in **Basic.** This disk is intended for your use and study of more "tricks"
Basic Tutor VII — More utility programs to use and for your study
Astronomy I — Introduction to the study of stars and constellations High resolution graphics is used to show the constellations you can find in the heavens. The first disk contains 6 programs
Astronomy II — A continuation of **Astronomy I,** that includes the viewing of the major constellations in the heavens with features shown Includes 6 programs
Physics I • Population • Number Race • Class Records Book • Inventory • Accounts • Conversions

All Disks $24 00 Partial listing, ask for our catalog. Pre-paid, add $1 50 Mailing/handling Guaranteed.

EDUCATIONAL COURSEWARE
3 Nappa Lane, Westport, CT 06880

CIRCLE 139 ON READER SERVICE CARD

ATARI 800 SOFTWARE!

>>>>> 3-DIMENSIONAL GRAPHICS PACKAGE <<<<<
HI-RES MULTICOLOR GRAPHICS TUTORIAL <<<<
for 8K to 48K!!.....ONLY- $29.95 +$1.50 P&H

> WUMPUS ADVENTURE- graphics & sound action<
very popular game- 16K......$14.95 +$.95 P&H

>> WUMPUS ADVENTURE 2- EXPANDED VERSION!! <<
(8K more than above!)...24K $17.95 +$.95 P&H

> DOWN THE TRENCH..Fast graphics & sound! <<
uses 1 joystick..8K or 16K..$15.95 +$.95 P&H

> SUBMARINE MINEFIELD- Navigator, for 1 or 2<
players with joysticks..8K...$8.95 +$.95 P&H

>> BATTLING TANKS........uses 2 joysticks <<
8K graphics game.......only- $9.95 +$.95 P&H

>> 3-D RED-BARON DOGFIGHT/FLIGHT SIMULATOR!<
requires 1 joystick....16K..$16.95 +$.95 P&H

> RANDOM SAMPLER #1- 4 graphics & game pro-
grams!!..8K and 16K.....only $9.95 +$.95 P&H

>> BASIC EDITOR.....only- $6.95 +$.95 P&H <<

>> HARDWARE!!: DIRECT SOUND OUTPUT CABLE! <<
With software!!......just- $17.95 +$1.75 P&H

> NOTE: ALL SOFTWARE WITH MANUAL & LISTINGS<
Programs on DISK for $5.00 extra PER ORDER.
Calif. residents please include 6% sales tax

>> SEND FOR YOUR FREE CATALOG TODAY FROM: <<
>>>> SEBREE'S COMPUTING..DEPT. 6T <<<<
456 GRANITE AVE.. MONROVIA CA. 91016
C.O.D.'s accepted. Call- 213-359-6092
WE CARRY FULL LINE OF BALLY ARCADE SOFTWARE!

CIRCLE 223 ON READER SERVICE CARD

EDUCATORS... Are You Using Microcomputers?

A major publishing company is seeking reviewers of CAI Software for grade levels K-12.

Reviewers should have experience with classroom use of one or more microcomputing systems (PET, Apple, TRS-80, etc.).

Write:
Dept. A
900 Sylvan Avenue
Englewood Cliffs, New Jersey
07632

CIRCLE 235 ON READER SERVICE CARD

NEW MUSIC SOFTWARE

TRS-80 LEVEL II (16K)

MINI-KEYS play 7 octaves like any keybd. instrument. 100's of speeds. **$12.95**

ReKord — hear prerecorded data play. Stars & Stripes Forever, Hallelujah! **$16.95**

MYTEE MUSIC — assemble over 3000 tones in a music score. E-Z to use. E-Z to edit. Whole notes thru 1/64, dotted & triplet. 100's of speed signatures. Instruction booklet makes program fun to use!

MYTEE MUSIC-1	good	$14.95
MYTEE MUSIC'2	better	$29.95

MYTEE MUSIC-1	good	$14.95
MYTEE MUSIC-2	better	$29.95
MYTEE MUSIC-3	best	$44.95

(add $5.00 for disc)

MYTEE MUSIC
P.O. Box 2432
Evansville, IN 47714

CIRCLE 227 ON READER SERVICE CARD

Now you can afford quality!

Scotch®
DISKETTES

SAVE 30% TO 45%!

Possibly the LOWEST prices you'll find on quality SCOTCH Brand Diskettes in sizes to fit all Mini/Micro Computers. One box or by the case, all orders are processed quickly and shipped per your instructions.

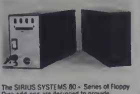
Call COLLECT to Order!
(805) 484-8146

QUALITY DATA SUPPLY
P.O. Box 236, Camarillo, CA 93010

Authorized Distributor
Information Processing Products **3M**

CIRCLE 243 ON READER SERVICE CARD

SIRIUS 80+
High Performance Low Cost Floppy Add-Ons!



The SIRIUS SYSTEMS 80+ Series of Floppy Disk add-ons are designed to provide unmatched versatility and performance for your TRS-80* Consisting of four different add-ons, there is a 80+ Series Floppy Disk Drive to meet your needs

COMMON CHARACTERISTICS

- *5ms track-to-track access time*
- *Auto-Eject*
- *180 day WARRANTY*
- *Exceptional speed stability - 11/2%*
- *Single/Double Density operation*
- *Mix any or all 80+ Series on the SS Standard cable*

SPECIFIC CHARACTERISTICS

The **SIRIUS 80+1** a single sided 40 track Drive Offering 5 more tracks than the Radio Shack model it cost $120 less Formatted data storage is 102K/204K Bytes Single/Double Density
SIRIUS 80+1 **$379.95**

The **SIRIUS 80+2** is a dual sided 80 track (40 per side) Disk Drive It appears to the TRS-80* as TWO 40 track drives yet **COST LESS THAN HALF THE PRICE!** Even greater savings result since data is recorded on both sides of the media instead of only a single side This unit may require the SS Standard cable Formatted data storage is 204K/408K Bytes Single/Double Density
SIRIUS 80+2 **$449.95**

The **SIRIUS 80+3** a single sided 80 track Drive Offering 2½ times the storage of a standard Radio Shack Disk Drive the 80+3 greatly reduces the need for diskettes correspondingly Additionally because of the increased storage and faster track-to-track access time, the 80+3 allows tremendously increased throughput for disk based programs! The 80+3 includes SIRIUS's TRAKS-PATCH on diskette (for use with 96 tpi drives) Formatted data storage is 204K/408K Bytes Single Double Density
SIRIUS 80+3 **$499.95**

The **SIRIUS 80+4** a dual sided 160 track (80 per side) 5¼" monster! The ultimate in state-of-the art 5¼" Floppy Disk Technology the 80+4 is seen by the TRS-80* as two single sided disk drives Thus, in terms of capacity one 80+4 is equivalent to 4⅔ standard Radio Shack drives — at a savings of over 73% (not to mention diskettes!!!) (With a double density converter the available memory is huge!) The 80+4 (a 96 tpi drive) includes TRAKS-PATCH on diskette and may require the SS Standard cable Formatted storage is 408K/816K Bytes Single/Double Density
SIRIUS 80+4 **$649.95**

All 80+ Series Floppy Disk add-ons operate at 5ms track to-track but are Expansion Interface limited to 12ms for the TRS-80*

**TRS-80 (c) of Tandy Corp*

ACCESSORIES

SS Standard 2 Drive Cable	**$29.95**
NEWDOS/80 Sophisticated Operating System for the TRS-80* from Apparat	**$149.95**

Save up to 10% *with these SIRIUS Packages!*

NEWDOS/80, SIRIUS 80+3 and Two Drive Cable	**$624.95**
NEWDOS/80 SIRIUS 80+4 and Two Drive Cable	**$749.95**
NEWDOS/80 Two (2) SIRIUS 80+3's Two Drive Cable	**$1080.95**
NEWDOS/80 Two (2) SIRIUS 80+4's Two Drive Cable	**$1349.95**

MPI 51/52 & 91/92 STATE-OF-THE-ART DISK DRIVES

- *Fast! 5ms track-to-track access*
- *Exclusive Pulley-Band Design*
- *Unique Door/Ejector Mechanism*
- *Reliable 11/2% Speed Stability*
- *Single/Double Density Operation*
- *Industry/ANSI Standard Interface*

MPI 51 (Single Head/40 tracks)
125K/250K Bytes Single/Double Density** **$259.95**

MPI52 (Dual Head/80 tracks (40/side))
250K/500K Bytes Single/Double Density** **$349.95**

MPI 91 (Single Head/80 tracks)
250K/500K Bytes Single/Double Density** **$399.95**

MPI 92 (Dual Head/160 tracks (80/side))
500K/1000K Bytes Single/Double Density** **$524.95**

MPI Technical Manual **$6.95**
**Unformatted data storage

QUME® DataTrak 8
8" Disk Drive
DOUBLE SIDED!
DOUBLE DENSITY!

$574.95

High performance Double Sided Disk 8" Disk Drive ■ Single or Double Density ■ Door Lock and Write Protect INCLUDED! ■ Negative DC Voltage not required ■ Low Power Operation

- *FAST! 3ms track-to-track access*
- *Low friction and minimum wear*
- *Superior Head Load Dynamics*

QUME DataTrak 8	**$574.95** (2/$549 ea)
QUME Technical Manual	**$6.95**
Connector Set #3 (AC DC & Card Edge)	**$10.95**
Connector Set #4 (AC and DC)	**$2.95**

TFORTH! -what it has to offer YOU!

TFORTH is a procedural FORTH type language which specifies a process rather than a desired result Designed to run on the TRS-80* TFORTH is a very powerful tool by itself or used in conjunction with Assembly Programming A rich set of WORDS come with TFORTH and many features considered as extra with other FORTH languages are standard with TFORTH These features include

- *Advanced Math Package*
- *Line Editor*
- *Macro Assembler*
- *Re Entrant Code*
- *Super Graphics Capabilities*
- *Sophisticated User Functions*
- *140 Page User's Manual*
- *Virtual memory*
- *Interpreter*
- *Compiler*
- *Produces CMD Files*
- *Expandable*
- *And many many other features*

TFORTH from SIRIUS comes on diskette complete for the TRS 80* with as little as 16K of memory and a single Disk Drive
TFORTH **$129.95**

SIRIUS SYSTEMS
7528 Oak Ridge Highway
Knoxville, Tennessee 37921

TO ORDER CALL (615) 693-6583
Phone Orders Accepted 9AM-7PM (EST) Mon-Fri

We accept **MC, VISA, AE, COD** (requires Certified Check, Cashier's Check or Cash) and **Checks** (personal checks require 14 days to clear) **SHIPPING AND HANDLING:** $7 00 per Floppy Disk Drive or 80 Module ■ 5% for other items (any excess will be refunded) ■ **Foreign Orders** add 10% for Shipping & Handling Payment in U.S currency ■ Tennessee residents add 6% Sales Tax ■ **VOLUME DISCOUNTS AVAILABLE**

CIRCLE 236 ON READER SERVICE CARD

PET, continued...

added at the end of the tokens table. This will not influence listings done on "old" ROM PETs.) To save space, the strings R$ and S$ are set to the Return and Space characters, respectively. Note that on the listing, S$="sp" really means: S$=" ". The why of this will be explained later.

After the tokens are in place, the ten special characters (cursor movements and the like) are read into the arrays C% and C$. The elaborate calculation in Line 63040 converts the upper case letters in N$ to their ASCII lower case values and sticks the result into C$. The result of this is the value in PET ASCII of a special character is found in C%() and its listing name in C$().

The program proper (ie, after all this setting up) starts in Line 63110 where a little line is asked for under PGM NAME:. . The OPEN then accesses the printer (my printer is IEEE device #4. If your printer is not on the IEEE bus, you will have to replace all of the PRINT#4 statements with a GOSUB to your printing routine.) We then print to the printer some carriage returns, the title line, and one more carriage return. Note that routine takes advantage of the PET's permissiveness on missing semicolons in PRINT statements. The CHR$(30) is a peculiarity of my Comprint printer, which wants to see this character to leave its automatic pagination mode. (My printerbeast will do pagination at very inappropriate times unless I explicity turn it off.) You can send any initialization for your printer, such as a form-feed, at this place.

Line 63120 computes the line number N from the line number stored after the Basic link pointer, and sets T to point at the first byte of the program line's text. P is then updated to look at the next line of program, and a test is applied to N.

VERY IMPORTANT! When you first use this program, change the end of Line 63120 to read: IF N=63000 THEN END . To list this program on my printer, I placed a REM at 63280 and changed Line 63120 to what you see here. 63000 is the first line in PRINTER LIST, and when it is seen, this test prevents the PRINTER LIST code from being listed.

Line 63130 prints the line number, N, and a space (here seen as "sp"). We then initialize Q, which is a flag for quote mode, and call Subroutine 63140 whose job is to actually list the contents of a line. Then a forced carriage-return via the PRINT#4 and on the next line via the GOTO.

At 63140, C is set to the current byte being scanned. A value of zero indicates the line is finished, so return. The job of Subroutine 63160 is to decide what to do with C. First, bump the pointer T up one, and then check C for quotation marks. (Quotes are CHR$(34).)

The expression Q=Q+1AND1 is an even/odd counter for the number of quotes found, with 0 for out-of-quotes mode and 1 for in-quote mode. At 63170 we decide if quotes are fashionable, and if so, skip to 63220.

Lines 63180-63210 handle non-quotes listing. First, the pi (π) token is checked for, and the sequence QS will print "pi" to the printer in Line 63230. Tokens larger than 203 (the largest token in non Basic 4 PETs) will force an asterisk as an indication that what is being listed is probably not a Basic program. In 63200, values below 128 are printed as they appear—things like TI$, for example. In 63210, we print the token's name by fetching the correct name from S$. (You can't scramble the tokens order in S$!)

Starting at 63220, we handle the in-quotes mode, which covers the string assignments, DATA and PRINT string literals. First we check for Space and Pi, and print these if found as "sp" or "pi". Values between 32 and 127 are printed as usual, as seen in Line 63240. If C exceeds 159, we know it to be a graphics character, and it is printed as the sequence (tilde) (PET Lower Case Equivalent). For example, the diamond (shift-Z) comes out as "~Z". The tilde warns of a shifted character.

All that's left to do are the special characters, checked for in Line 63260. Line 63270 provides a graceful exit if the special character isn't recognized.

One difference from my usual ways of typing a program is that the tilde will indicate a shifted character instead of an underline. As my typewriter does not have a tilde, I will still underline any short bits of program in the text of this column.

If you want to use some other representation of the PET's special characters, the change is easily made. Simply change the names in Line 63100, for example,

```
63100 DATA 147,CLEAR,19,HOME, .... (etc)
```

In line 63260, surround the CS() with some brackets, ie,

```
63260 FORJ=0TO9:IFC=C%(J)THENPRINT#4,
      "["C$(J)"]";:RETURN
```

We will now see cursor movements as (CLEAR), (HOME) and so on.

As a challenge, do a "look-ahead" in PROGRAM LIST which 1) counts the number of repeated characters for cursor movements and spaces only—so we see something like:

6-right 5-space

appearing in the listing. When spaces are counted, if they appear between letters, like "HELLO OUT THERE" the listing gives us:

"HELLO OUT 2-space THERE"

I am sure this will keep you busy for an evening. Send me the results, I can use it!

If You Really Want to Know About Your Program

Benson Greene, 210 Fifth Ave, New York, NY 10010 sent me a program, *Proganal*, which reads a Basic program from tape (There's a *Diskanal* for disk systems too.), generates a listing and then produces several pages of reference information, including the variable names with the line numbers where these variables are found, a usage chart of the various Basic tokens, the line numbers where branches (jumps, GOSUBs and GOTOs) exist, and a summary of several statistics such as number of program lines, number of statements, etc. Benson will supply a tape of this program for $10.00, if you also provide a SASE.

My feelings toward this program are mixed. First, it is an exercise in looking at the various parts of Basic program, and the reports are useful if you are looking at a program after a long time or at someone else's program. (Some programmers I know will do things like RENUMBER 1,1 just to be nice!) Second, *Proganal* takes a long time to run, generally 30 minutes or more for a program of any complexity. This is good night-time work for your PET.

On the other hand, *Proganal* isn't to be used much during the development of a program, due to the numerous changes that occur in most program debugging sessions. There is also a matter of skill—those of you who feel confident in Basic will probably not find many uses for *Proganal*. In any case, *Proganal* is of interest for programwriters, particularly in thinking of how *Proganal's* functions might be done or improved.

There is one problem in computing which might be tackled with methods similar to those used in PROGRAM LIST and *Proganal*. Given a program with no INPUT or GET statements, determine if the program will stop executing. For example,

```
10 END
```

will definitely halt, and

```
10 GOTO 10
```

will definitely not halt.

Now for a curve ball: When testing your program, use the same program for analysis........ (those of you who know about this problem and its consequences are warned that this presentation is to encourage the rest of my readers to do some mental exercises.)

The MONJANA/1 CBM ROM

If you have no interest in 6502 machine language, skip this review. The Monjana/1 is a 2k ROM which is installed in the $9000 socket of "new" ROM PETs, and provides several facilities which assist the machine language programmer. (Available from Elcomp Publishing, 3873L Schaefer Ave, Chino, CA 91710 for $98.00. Calif residents add 6% tax.) Figure 1 shows a printer dump of the Monjana's activity through several commands. (An interesting feature of this ROM is that its output can be sent to the

PET, continued...

printer for later use, akin to CMD in Basic. This very useful function is one I haven't seen in several other monitors and assemblers.)

The first part is M (memory from $9000 to $9060—I might as well take a look at the ROM's code! The second part is a D (disassemble) of the same part of memory. The Monjana conveniently remembered the $9000 to $9060 from my M command, so only a carriage-return was needed after entering the D commond. (Monjana commands first accept the letter, and then write on the screen to help you. For example, after entering M, we see: •M 0000-0000 with the cursor in the first 0.) The third part is after N (note) which permits comments on the printer.

The fourth part is a trace, here showing the jump forced by the PET if started in address $0000. At $D123 lives the PET's Monitor, and we see the beginning of the Monitor's operation.

An unusual (though necessarily limited) feature is that of relocation and relinking. The relocation will change all jumps and JSR's to be compatible with the new location for a program, and the linking will patch any local addresses (such as in LDA) which refer to locations in the moved area. Provided you avoid complex tricks like setting the stack and then using RTS to do a jump for you, the relocation should work reasonably well.

One negative aspect of the Monjana is the $100 price tag. This seems a bit steep to me, as the Programmer's Toolkit (of at least as much use) runs from $50 to $80.

A Big Keyboard If You Need It

Though the majority of PETs now have the full size keyboard, there are still many machines which lack this grace. Century Research & Marketing will provide you with a full sized keyboard for $139.95. (Century Research & Marketing, 4815 West 77th Street, Minneapolis, Minn. 55435.)

I installed the keyboard on my PET (all of 3 minutes to do) and have used it regularly for 3 months. Once you are used to the layout—some of the keys, such as RUN/STOP are in slightly different positions— the speedup possible with a full keyboard is quite valuable. On the numeric pad, the ?, Space, Return and Shift keys are duplicated to assist you in data entry. (The ? for PRINT in calculator mode.)

The plastic case of the keyboard looks flimsy, but I find that it stands up well. The special function keys (cursor movments, Shift, and so on) are in blue for rapid location. The keyboard is less bulky than the other unit available (Skyles Big Kbd, which is no longer in production, though a few are still in stock. Call 415 965-1735)

So, if you need it, get it—Manufacturers of PET keyboards face a declining market, and soon your choice will be limited to building a "homebrew" unit. (I've done that one too....)

Hangmath Returns

In the May 1980 column I described a program, *Hangmath*, and some improvements to make the program more playable and enjoyable. The job wasn't complete, however, and Paul Agosta sent me an improved version which provides a report of past guesses and (I'm impressed!) draws the gallows, etc. on the right side of the screen. Program 2 shows the improved version of *Hangmath* for your enjoyment. (And as an exercise in reading the output of PRINTER LIST.)

Do try it out—the improvements are worth seeing. Deliberately lose a game to see the animation—simple, and effective. Perhaps Creative Computing Software would be interested in this? □

```
HANGMATH AS IMPROVED BY PAUL AGOSTA

 10  REM •• SEE TITLE PAGE FOR CREDITS ••
 20  REM ••   USE ONLY  NOT FOR SALE   ••
 40  CLR:GOSUB2000
 50  XX$="up sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp s
p sp sp sp sp sp sp sp sp sp sp sp sp sp sp "
 60  S$="hm dn dn dn dn dn dn dn dn dn dn "
 70  DIM A(4,S),B(4,S),C(4,S),E$(4,S),HM$(12),P$(13)
 72  FORJ=1TO12:READHM$(J):NEXT
 73  DATA "HEAD","UPPERsp BODY","LOWERsp BODY","LEFTsp ARM","RIGHTsp ARM","LEFTs
p LEG"
 74  DATA "RIGHTsp LEG","LEFTsp HAND","RIGHTsp HAND","LEFTsp FOOT","RIGHTsp FOOT
"
 75  DATA "sp sp sp sp sp sp sp •••••sp YOU'REsp HUNG!!sp •••••"
 76  P$(0)="lft lft lft lft lft lft lft lft lft lft lft lft rvs ~•~•~•~1~•~•~•~•
~•~•~•~•lft lft lft lft lft lft lft lft lft lft lft lft dn sp sp HANGMATHsp sp l
ft lft lft lft lft lft lft lft lft up "
 77  P$(0)=P$(0)+"up ~]up lft ~]up lft ~]up lft ~]up lft ~]up lft ~]up lft ~]up
lft ~]up lft ~0~•~•~•~•~.off dn lft ~]"
 78  P$(1)="lft lft lft lft up up up up up up up ~W"
 79  P$(2)="lft lft lft lft up up up up up up ~["
 80  P$(3)="lft lft lft lft up up up up up ~1"
 81  P$(4)="lft lft lft lft lft up up up up up up ~-"
 82  P$(5)="lft lft lft up up up up up up ~="
 83  P$(6)="lft lft lft lft lft up up up up up ~U"
 84  P$(7)="lft lft lft up up up up up ~I"
 85  P$(8)="lft lft lft lft lft up up up up up up up ~;"
 86  P$(9)="lft lft lft up up up up up up up ~,"
 87  P$(10)="lft lft lft lft lft up up up up ~>"
 88  P$(11)="lft lft lft up up up up ~<"
 89  P$(12)="lft lft lft lft lft up up up up up up ~,~]~;dn lft lft lft ~-~[~=dn
 lft lft lft ~U~1~Idn lft lft lft ~<sp ~>dn dn dn rt rt "
 90  P$(13)="lft lft lft lft lft up up up up up up up sp rt sp dn lft lft lft sp
 ~]sp dn lft lft lft sp sp sp dn lft lft lft sp sp sp dn lft lft lft sp sp sp dn
 dn dn rt rt ":POKE59468,12
 91  PRINT"clr sp SETTINGsp UPsp ...."
 92  FORB=0TO10:K(B)=-1:O$(B)="":CK$(B)="":NEXT
 94  PRINTS$;"dn dn dn dn dn dn "
 100  W=0:F$="":R1=RND(-TI)
 110  C$="ABCDEFGHIJ"
 120  R=INT(LEN(C$)*RND(1)+1)
 125  N$=MID$(C$,R,1):F$=F$+N$
 130  IFR>1THENT$=LEFT$(C$,R-1):GOTO150
 140  T$=""
 150  C$=T$+MID$(C$,R+1)
 160  IFLEN(C$)>0THEN120
 170  IFLEN(F$)<>10THEN100
 210  N(0)=100+INT(RND(1)*900)
 220  N(1)=10+INT(RND(1)*90)
 230  N(2)=N(0)*(N(1)-10*INT(N(1)/10))
 240  N(3)=N(0)*INT(N(1)/10)
 250  N(4)=N(0)*N(1)
 255  IF10*INT(N(1)/10)=N(1)THEN220
 260  FORX=0TO4:C=0:FORY=1TOS
 270  A(X,Y)=INT(N(X)/INT(10↑(S-Y)))
 272  B(X,Y)=A(X,Y-1)
 274  C(X,Y)=A(X,Y)-10*B(X,Y):E$(X,Y)=MID$(F$,C(X,Y)+1,1):C=C+C(X,Y)
 280  IFC=0THENE$(X,Y)=""
 290  NEXT:L(X)=LEN(STR$(N(X))):NEXT
 400  PRINT"hm HANGMATHsp sp sp sp sp sp sp sp dn •up lft ";
 402  FORX=0TO4:FORY=1TOS:T=20
 404  IFX=3THENT=10
 410  PRINTTAB(T-L(X));E$(X,Y);"sp ";
 420  NEXT:IFX=1ORX=3THENPRINT:PRINTTAB(13);"~•~•~•~•~•~•~•~•~•~•~•up "
 430  PRINT:NEXT:PRINT
 440  IFW=0THENGOSUB5000
 500  T=1:GOSUB3200:IFFTHENPRINT"lft rvs QUIT":W=12:GOSUB5000:GOTO1000
 501  FORJ=0TO10:IFZ$=CK$(J)THENM$=Z$+"sp ="+STR$(J)+",sp BOZO":GOSUB3500:GOTO50
0
 503  NEXT:IFN=K(N)THENM$=MID$(F$,N+1,1)+"sp ="+STR$(N)+",sp DUMBO":GOSUB3500:GO
TO500
```

Our State-of-the-Art thinking gave you SARGON. But chess isn't all we think about!

SARGON II (Spracklen) The Champ of champs ". an excellent program which will provide a true challenge for many players. Save your money and buy SARGON II " '80 *Software Critique* *03403, TRS 80 Level II; 03404, Apple II; 03410, OSI C1P; 03440, OSI C4P; 03401, PET, each tape $29 95 03408, TRS 80 Level II Disk, 03409, Apple II Disk; 03414, OSI C1P Disk, 03444, OSI C4P Disk, 03484, C8P Disk, each $34 95*

BLACKJACK MASTER: A Simulator/Tutor Game (Wazaney) A serious game that performs complex simulations and evaluations of playing and betting strategies 05303, *TRS 80 Level II tape*, $24 95; 05308, *TRS 80 Disk Version*, $29 95

REVERSAL (Spracklen)
Winner of the software division of the *First International Man Machine OTHELLO™ Tournament*, this version of the 200 year old game Reversi, features 27 levels of play and high resolution color graphics *07004, APPLE II tape, $29 95; 07009, APPLE II Disk, $34 95*

APPLESOFT UTILITY PROGRAMS (Gilder)
Increase your BASIC programming speed and flexibility Contains 9 useful subroutines 1 REM Writer 2. PRINT Writer 3 POKE Writer 4 Hexadecimal Decimal Converter 5 Line Counter 6 Renumber 7. Append 8 Byte Counter 9. Slow List Stop List *03504, Apple II tape, $29 95*

FLASH & CRASH SOUND EFFECTS
(American Micro Products)
A collection of 18 subroutines that can be incorporated into your own programs to produce sound effects with the American Micro Products music board. Included are: Train, Explosion, Phaser, Chimes, Sirens, Jet and 12 others *08709, APPLE II Disk, $39 95*

6502 DISASSEMBLER (Stamm)
Produce assembly language source files with labeled subroutines and references from programs already in memory It is compatible with Hayden's ASSEMBLY LANGUAGE DEVELOPMENT SYSTEM *08609, APPLE II Disk, $34 95*

PSEUDODISK (Neuschatz)
This money-saving program simulates a disk memory system for Integer BASIC programs It allows multiple programs in memory at the same time which can be run from a catalog *04804, APPLE II tape, $24 95*

LINE & VARIABLE CROSS REFERENCE GENERATOR (Johnson) Provides a cross reference of line numbers and variable names *07301, PET tope, $16 95*

DISK CATALOGER (LeBar)
Automatically maintains a cross-reference listing of all your programs, their location by disk number, their function and use Catalogs, lists and sorts programs. *05203, TRS 80 Level II tope, $16 95; 05208, TRS 80 Level II Disk, $21.95*

APPLE™ ASSEMBLY LANGUAGE DEVELOPMENT SYSTEM: An Assembler/Editor/Formatter (Lutus) Write and modify your machine language programs quickly and easily. *04609, Apple II Disk Version, $39 95*

SUPER APPLE™ BASIC (Lutus) A structured BASIC that compiles into an optimized Applesoft or Integer BASIC program. *05409, Apple II Disk, $39 95*

MAILING LIST (Tru Data Software) Lists addresses, prints labels, allows for alterations and deletions, and has the capacity to make duplicate data file disks Can only be used with version 1.5 *05713, Heath tape, $49 95*

FINPLAN: A Financial Planning Program for Small Business (Montgomery) Allows you to enter data from a balance sheet into the program, to make assumptions about future growth of business, and to have the computer project results for up to a five year period based on those assumptions And if you change any data, the program revises all resulting data automatically. The disk version can only be used with TRSDOS Version 2 3 *05103, TRS 80 Level II tape, $69 95, 05108, TRS 80 Level II Disk Version, $74 95*

DATA MANAGER: A Data Base Management System and Mailing List (Lutus) Store information on a floppy disk, and retrieve it quickly and easily by specific names, or by category. *04909, Apple II Disk Version, $49 95*

MCAP: A Microcomputer Circuit Analysis Program (Savon) Performs a linear voltage, impedance, or transfer impedance analysis of an electronic circuit. *04501, PET; 04503, TRS 80 Level II, 04504, Apple II, each tape $24 95, 04513, Heathkit/Zenith Disk, $29 95*

MICROCOMPUTER AIDED DESIGN OF ACTIVE FILTERS (Gilder) Eight programs that simplify the design of active filters and will calculate the component values needed for various bandpass, low-pass, and notch type filters *01401, PET, 01403, TRS 80 Level II, 01404, Apple II, 01407, Heath, each tape $16.95, 01413, Heathkit Zenith Disk Version, $21 95*

DISK CERTIFIER AND COPIER (Jacc Inc)
A handy utility program that certifies the acceptability of blank diskettes and rejects those with flaws It also includes a fast machine language disk copying program that will work on single and dual drive systems *07809, APPLE II Disk, $19 95*

SONGS IN THE KEY OF APPLE (Lopatin)
Allows you to see and hear your favorite tunes, pre programmed tunes or music you create (up to 200 notes, including rests, per musical piece) *03304, Apple II tape, $10 95*

HOW TO BUILD A COMPUTER-CONTROLLED ROBOT (Loofbourrow) Contains 5 control programs that consist of Joystick Control Program, Self Direction Program, Impact Sensor Control Routine, and more *00100, KIM 1 tape, $14 95* Should be used with text **HOW TO BUILD A COMPUTER-CONTROLLED ROBOT**, 5681 8, $9 75

Apple is a trademark of Apple Computer Company Inc and is not affiliated with Hayden Book Company Inc

ORDERING INFORMATION

Hayden Book Company, Inc.
50 Essex Street, Rochelle Park, NJ 07662

Send me the software checked below A check or money order is enclosed I understand that Hayden pays shipping and handling costs and that I can return any disk or tape within 10 days if it is defective or I am dissatisfied with it for any reason Residents of NJ and CA must add sales tax Offer good in US only

☐ 00100	☐ 02404	☐ 03401	☐ 03414	☐ 04503	☐ 05103	☐ 07009
☐ 01401	☐ 02501	☐ 03403	☐ 03440	☐ 04504	☐ 05108	☐ 07301
☐ 01403	☐ 02503	☐ 03404	☐ 03444	☐ 04513	☐ 05203	☐ 07809
☐ 01404	☐ 02601	☐ 03408	☐ 03484	☐ 04609	☐ 05208	☐ 08609
☐ 01407	☐ 02701	☐ 03409	☐ 03504	☐ 04804	☐ 05308	☐ 08709
☐ 01413	☐ 03304	☐ 03410	☐ 04501	☐ 04909	☐ 05409	☐ 5681 8

Name ____________________

Address ____________________

City/State/Zip ____________________ CC 1 81

Name of individual ordering must be filled in

Call Toll Free

24 hours a day,
(1 800 821-3777, ext. 302)* TO CHARGE YOUR ORDER TO Master Card or Visa. Minimum order is $10.00; customer pays postage and handling.
From Missouri call (1-800 892 7655, ext. 302)

Hayden Book Company, Inc.

50 Essex Street, Rochelle Park, NJ 07662

CIRCLE 216 ON READER SERVICE CARD

Memory Dump

```
M:9000-9060    0  1  2  3  4  5  6  7
W:9000-9007   A0 01 D0 08 A0 90 84 93
W:9008-900F   A0 00 84 92 58 D8 20 DB
W:9010-9017   94 90 F0 14 AA 68 95 20
W:9018-901F   E8 E0 07 90 F8 A5 25 D0
W:9020-9027   02 C6 26 C6 25 88 F0 F5
W:9028-902F   BA 86 20 A5 C6 F0 05 A9
W:9030-9037   0D 20 D2 FF 98 D0 2C 2C
W:9036-903F   A0 21 20 71 96 A0 12 A9
W:9040-9047   02 20 E7 F1 A2 60 9A D9
W:9048-904F   BA 97 F0 07 88 10 F8 A0
W:9050-9057   29 D0 E7 AA 86 1F B9 D1
W:9058-905F   97 48 B9 E8 97 48 A9 17
W:9060-9067   85 27 60 A2 42 20 5B 96
```

Disassembly

```
D:9000-9060    1  2  3     MNC-CODE
I:9000        A0 01        LDY =$01
I:9002        D0 08        BNE $900C
I:9004        A0 90        LDY =$90
I:9006        84 93        STY $93
I:9008        A0 00        LDY =$00
I:900A        84 92        STY $92
I:900C        58           CLI
I:900D        D8           CLD
I:900E        20 DB 94     JSR $94DB
I:9011        90           TYA
I:9012        F0 14        BEO $9028
I:9014        AA           TAX
I:9015        68           PLA
I:9016        95 20        STA $20,X
I:9018        E8           INX
I:9019        E0 07        CPX =$07
I:901B        90 F8        BCC $9015
I:901D        A5 25        LDA $25
I:901F        D0 02        BNE $9023
I:9021        C6 26        DEC $26
I:9023        C6 25        D[illegible] $25
I:9025        88           DEY
I:9026        F0 F5        BEO $901D
I:9028        BA           TSX
I:9029        86 20        STX $20
I:902B        A5 C6        LDA $C6
I:902D        F0 05        BEO $9034
I:902F        A9 0D        LDA =$0D
I:9031        20 D2 FF     JSR $FFD2
I:9034        98           TYA
```

Note To Printer

```
I:905E        A9 17        LDA =$17
I:9060        85 27        STA $27


N:THIRD MONJANA EXAMPLE
```

Trace

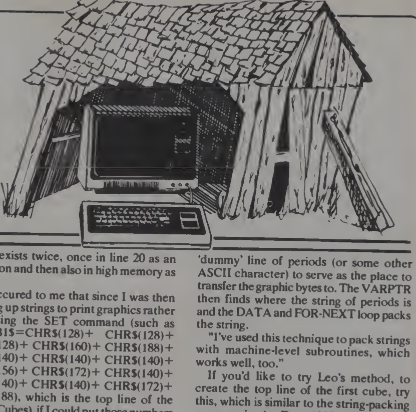
```
T:<PC> SR AC XR YR SP   MNC-CODE
E:0000 00 00 00 00 F8   JMP $D123
E:D123 00 00 00 00 F8   LDX =$35
E:D125 30 00 35 00 F8   JMP $C357
E:C357 30 00 35 00 F8   LSR $0D
E:C359 32 00 35 00 F8   LDA $0E
E:C35B 32 00 35 00 F8   BEQ $C364
E:C364 32 00 35 00 F8   JSR $C9E2
E:C9E2 32 00 35 00 F6   LDA =$0D
E:C9E4 30 0D 35 00 F6   JSR $CA45
E:CA45 30 0D 35 00 F4   BIT $0D
E:CA47 32 0D 35 00 F4   BMI $CA4C
E:CA49 32 0D 35 00 F4   JSR $FFD2
E:FFD2 32 0D 35 00 F2   JMP $F232
E:F232 32 0D 35 00 F2   PHA
E:F233 32 0D 35 00 F1   LDA $B0
E:F235 30 03 35 00 F1   CMP =$03
E:F237 33 03 35 00 F1   BNE $F23D
E:F239 33 03 35 00 F1   PLA
```

```
504  IF2$=O$(N)THENM$="YOUsp GUESSEDsp THATsp BEFORE":GOSUB3500:GOTO500
520  IF2$=MID$(F$,N+1,1)THEN600
530  W=W+1:O$(N)=2$:GOSUB5000
531  IFW=1THEN PRINT"up up sp sp sp sp sp sp sp sp sp sp sp sp YOURsp GUESSESdn
lft lft lft lft lft lft lft lft lft lft lft lft ~●~●~●~●~●~●~●~●~●~●~●~●"
532  PRINTTAB(W*10-10);Z$;"sp =";N;
533  IFW=12THEN1000
535  GOTO500
N1=50:FORX=0TO4:FORY=1TO5
610  IFE$(X,Y)=2$THENE$(X,Y)="lft "+STR$(N):CK$(N)=2$:K(N)=N
620  IFE$(X,Y)=""THEN680
630  N=ASC(E$(X,Y))
632  IFN=157THENM=0
640  IFN>N1THENN1=M
680  NEXTY:NEXTX
682  IFN1<60THEN700
690  PRINT:GOTO400
700  PRINT"dn sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp ●●●●●●●●●●sp sp sp sp s
p sp sp sp sp sp sp sp sp dn "
702  PRINT"sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp YOUsp GOTsp ITsp sp sp sp
sp sp sp sp sp sp sp sp sp sp ":GOTO1042
710  PRINT"up sp sp sp sp sp sp sp sp sp sp YOUsp HAD"W"MISTAKESsp sp sp sp sp
sp sp sp sp sp "
720  T1=T1+W:U=U+1:T2=T2+W*W
730  PRINT"dn sp sp sp sp sp AVERAGEsp AFTERsp "U"GAME(S)sp IS"INT((T1/U)+.5)"l
ft sp sp sp dn ";XX$
750  PRINT"sp sp sp sp sp sp sp sp sp sp sp sp ANOTHERsp GAMEsp ?sp ";
755  GOSUB3000
760  IFA$="Y"THEN91
764  IFA$<>"N"THEN755
770  PRINTS$;"dn SOsp LONGsp !!sp sp sp sp sp sp sp sp ":END

1000  REM SOLUTION SUBROUTINE
1001  PRINT"hm rvs SOLUTIONoff sp sp sp sp sp sp rt rt ";
1002  FORX=0TO4:FORY=1TO5:T=20
1003  IFX=3THENT=18
1004  E$(X,Y)="lft "+5TR$(C(X,Y))
1005  IFC(X,Y)=0ANDX<2ANDY<4THENE$(X,Y)=""
1006  IFC(X,Y)=0ANDX=2ANDY<3THENE$(X,Y)=""
1007  IFC(X,Y)=0ANDX=3ANDY<3THENE$(X,Y)=""
1008  IFC(X,Y)=0ANDX=4ANDY=1THENE$(X,Y)=""
1010  PRINTTAB(T-L(X));E$(X,Y);"sp ";
1020  NEXT:IFX=1ORX=3THENPRINT:PRINTTAB(13);"~●~●~●~●~●~●~●~●~●~●~●up "
1030  PRINT:NEXT:PRINT
1038  PRINT"hm dn dn dn dn dn dn dn dn dn dn dn dn dn dn sp sp sp sp ";
1040  PRINT"●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●sp sp sp sp sp sp sp sp sp sp sp
sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp
1041  PRINT"sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp "
sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp sp YOUsp MISSEDsp ITsp sp sp sp
1042  FORJ=1TO4:PRINT"dn ";XX$;:NEXT:PRINT"up up up "
1050  GOTO710
2000  REM INSTRUCTIONS
2010  POKE59468,14:PRINT"clr ";TAB(15);"dn dn rvs sp ~H~A~N~G~M~A~T~Hsp off dn
dn "
2011  PRINT"sp sp sp sp ~ORIGINALsp ~AUTHORS:sp ~PHILsp ~FELDMANdn lft lft lft
lft lft lft lft lft ~TOMsp ~RUGG"
2012  PRINT"dn sp sp sp sp ~APRIL,sp 1977sp sp ~KILOBAUD,sp PG.sp 112
2013  PRINT"dn sp sp sp sp ~MODIFICATIONSsp BY:sp ~J.sp ~R.sp ~ROSSUMsp "
2014  PRINT"sp sp sp sp ~MOREsp CHANGESsp sp BY:sp sp ~GREGORYsp ~YOBdn lft lft
lft lft lft lft lft lft lft lft lft ~PAULsp ~AGOSTAdn "
2020  PRINT"dn dn sp sp sp sp ~DOsp YOUsp NEEDsp INSTRUCTIONSsp ?sp ";:GOSUB300
0
2030  IFA$<>"Y"THENRETURN
2040  PRINT"clr sp sp sp sp sp sp sp sp ~H~A~N~G~M~A~T~H~ ~I~N~S~T~R~U~C~T~I~O~
N~Ssp "
2050  PRINT"dn dn sp sp ~HANGMATHsp WILLsp PRESENTSsp YOUsp WITHsp A"
2060  PRINT"MULTIPLICATIONsp PROBLEMsp WHICHsp HASsp THE
2070  PRINT"DIGITSsp REPLACEDsp BYsp LETTERS.
2080  PRINT"dn ~FORsp EXANPLE:dn "
2090  PRINT"sp sp sp sp sp sp sp sp Asp Bsp Fsp sp sp sp sp sp sp sp sp sp 4
sp 5sp 6
2100  PRINT"sp sp sp sp sp sp sp ●sp sp Esp Dsp sp sp sp sp sp sp sp sp sp ●sp
sp 3sp 2
2110  PRINT"sp sp sp sp sp sp sp ~●~●~●~●~●~●sp sp sp sp sp sp sp sp sp sp ~●~●
~●~●~●~●
2120  PRINT"sp sp sp sp sp sp sp sp Hsp Csp Dsp sp sp sp sp sp sp sp sp sp 9
sp 1sp 2
2130  PRINT"sp sp sp sp Csp Esp Fsp Isp sp sp sp sp sp sp sp 1sp 3sp 6sp 8
2140  PRINT"sp sp sp sp ~●~●~●~●~●~●~●~●~●sp sp sp sp sp sp sp ~●~●~●~●~●~●~●~●
~●
2150  PRINT"sp sp sp sp Csp Asp Bsp Hsp Dsp sp sp sp sp sp sp 1sp 4sp 5sp 9sp 2
2160  PRINT"dn dn sp sp sp ~HEREsp THEsp LETTERsp '~A'sp ISsp THEsp DIGITsp 4
2170  PRINT"THEsp LETTERsp '~H'sp ISsp THEsp DIGITsp 9sp ANDsp SOsp ON.":GOSUB3
100
2200  PRINT"clr dn dn sp sp sp sp sp sp -~I~N~S~T~R~U~C~T~I~O~N~S~ ~C~O~N~T~I~N
~U~E~D-
2210  PRINT"dn dn sp sp ~TOsp ENTERsp Asp GUESS,sp JUSTsp TYPEsp THE"
2220  PRINT"LETTERsp ANDsp NUMBER.sp sp ~FORsp EXAMPLE,sp '~A4'
```

```
2230  PRINT"ANDsp '4~A' sp WILLsp BOTHsp WORKsp FORsp MATCHING"
2240  PRINT"THEsp LETTERsp '~A' sp WITHsp THEsp DIGITsp '4'.
2250  PRINT"dn sp sp ~YOUsp WILLsp BEsp GIVENsp 12sp CHANCESsp TOsp SOLVE
2260  PRINT"THEsp PUZZLEsp BEFOREsp THEsp SOLUTIONsp ISsp sp sp sp sp sp sp DIS
PLAYED."
2270  PRINT"dn sp sp ~IFsp YOUsp PRESSsp 'Q',sp THEsp SOLUTIONsp WILLsp sp sp B
Esp GIVEN.":GOSUB3100:RETURN
3000  REM * INPUT CHARACTER *
3010  PRINT"rvs ~?lft ";:FORJ=1TO350:NEXT
3020  PRINT"off ~?lft ";:FORJ=1TO350:NEXT
3030  GETA$:IFA$=""THEN3010
3040  RETURN
3100  PRINT"dn dn sp sp sp sp sp sp PRESSsp rvs SPACEsp BARoff sp TOsp CONTINUE
sp ";
3110  GOSUB3000:RETURN
3200  REM GUESS ENTRY
3215  PRINTS$
3220  F=0:Z$="@":N=-1
3230  PRINT"YOURsp GUESS:sp sp sp lft lft ";
3240  FORK=1TO2
3250  GOSUB3000:IFA$=CHR$(13)ORA$=CHR$(17)ORA$=CHR$(19)ORA$=CHR$(29)THEN3250
3255  PRINTA$;
3260  IFA$="Q"THEN F=1:RETURN
3270  REM FORM Z$,N
3280  IFA$>"@"THENZ$=A$:GOTO3300
3290  N=ASC(A$)-48
3300  NEXTK
3310  REM CHECK VALIDITY
3320  IFN<0ORN>9THEN3350
3330  IFZ$<"A"ORZ$>"J"THEN3350
3340  PRINT:RETURN
3350  PRINT:PRINT"up BADsp NUMBERsp ORsp LETTER"
3360  FORJ=1TO800:NEXT
3370  PRINTXX$
3380  PRINT"up ";:GOTO3220
3500  REM TEMP MESSAGE DISPLAY
3520  PRINTS$;"dn dn dn ";M$
3530  FORJ=1TOT*1000:NEXT
3540  PRINTXX$
3550  PRINT"dn dn "
3560  RETURN
```

"And just how big is this new computer?"

```
5000  REM MISS SUBROUTINE
5001  IFW<>12THEN5005
5002  PRINTS$;
5003  FORJ=1TO3:PRINTP$(13);"dn ";:PRINTP$(12);:FORK=1TO25:NEXTK
5004  NEXT:T=5:M$=HM$(W):GOTO5050
5005  PRINTS$;P$(W):T=3
5006  IFW=0THENRETURN
5010  M$="**sp WRONGsp **sp sp DRAWsp THEsp "+HM$(W)
5050  GOSUB3500:RETURN
```



TRS-80 Strings

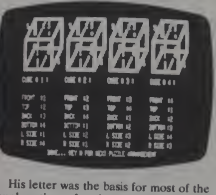
Stephen B. Gray

For column number 26, let's take another look at string packing with some comments by top programmer Leo Christopherson, two programs for renumbering program lines, a software directory, Radio Shack's Astrology program, five moves that will beat Microchess, and the tale of a vanishing van.

String Packing Revisited

In the Dec. 1980 column, we discussed how to create screen graphics using string packing, which was discovered by Leo Christopherson. String packing may have been simultaneously discovered by several other program writers, who will undoubtedly be heard from.

Back when I first checked out Leo's Cubes program (Aug. 1980, p 162), which is the computer version of Parker Brothers' Instant Insanity game that uses four colored plastic cubes, I asked Leo how he created the four cubes on the screen.



His letter was the basis for most of the explanation of string packing in that December column. Because that letter explains how he discovered the priniciple, here's most of it:

"Back when I was finishing Android Nim with sound, and Snake Eggs, I noticed that strings that were defined in the Basic text (such as 20 A$="1234") were not stored in high memory. But when one builds a string (such as 20 A$="1"+"2"+"3"+"4" or 20 A$=CHR$(49)+ CHR$(50)+ CHR$(51)+ CHR$(52)), then the actual string of "1234" is way up in high memory. It, in effect, exists twice, once in line 20 as an operation and then also in high memory as "1234".

"It occured to me that since I was then building up strings to print graphics rather than using the SET command (such as 1000 B1$=CHR$(128)+ CHR$(128)+ CHR$(128)+ CHR$(160)+ CHR$(188)+ CHR$(140)+ CHR$(140)+ CHR$(140)+ CHR$(156)+ CHR$(172)+ CHR$(140)+ CHR$(140)+ CHR$(140)+ CHR$(172)+ CHR$(188), which is the top line of the cube in Cubes), if I could put those numbers (128, 128, 128, 160, 188, and so on) in a string located in the Basic text, this would save room and the string would not have to be constructed in high memory each time the program was run.

"Well, it turns out that the VARPTR command will find the start of a string anywhere in memory. So I entered a string in this form: 1000 B1$= "..............." (15 periods).

"Then I used VARPTR to find the start of B1$: N = PEEK (VARPTR (B1$) + 2) * 256 + PEEK (VARPTR (B1$) +1).

"Then I set up a data line as follows: DATA 128, 128, 160, 188, etc.

"This was followed by a FOR-NEXT loop: RESTORE: FOR X = 0 TO 14: read D: POKE N + X, D: NEXT X: STOP.

"Now, after running this, the string looks like it does when you list line 1000 of Cubes, (B1$="ENDENDENDOUT... and so on).

"So you see, the graphics are being printed on the screen as a string, not with set commands. Those token words occur since Microsoft Basic uses the numbers 128 to 255 as tokens for their commands when listing programs. I suppose they never intended for strings with bytes of higher than ASCII numbers in them to appear in the Basic text. But it works! The CHR$(128) or a blank, appears as END in a listing. So, from my point of view, the appearance of those token words is of no concern to the purpose of string packing.

"So, to review, to use this technique in a program, you first have to plan out your graphics grames as a series of lines of graphic CHR$( ) numbers. Then set up a 'dummy' line of periods (or some other ASCII character) to serve as the place to transfer the graphic bytes to. The VARPTR then finds where the string of periods is and the DATA and FOR-NEXT loop packs the string.

"I've used this technique to pack strings with machine-level subroutines, which works well, too."

If you'd like to try Leo's method, to create the top line of the first cube, try this, which is similar to the string-packing program in the Dec. 1980 column, with some slight changes to slim it down for this column width:

```
100 CLS
1000 B1$="..............."
1050 L=PEEK(VARPTR(B1$)+2)*256
1060 R=PEEK(VARPTR(B1$)+1)
1070 N=L+R
1100 DATA 128,128,128,160,188
1110 DATA 140,140,140,156,172
1120 DATA 140,140,140,172,188
1160 FOR X=0 TO 14
1170 READ D
1180 POKE N+X,D
1190 NEXT X
1200 PRINT B1$
1300 GOTO 1300
```

Incidentally, once you've run this program, you can delete all the lines except 1000 and 1200, and you'll still get the top line of the cube if you run just those two lines.

A Misunderstanding

In my review of Leo's Cubes and Snake Eggs programs (which are published at $9.95 and $14.95, respectively, by 80-US Software, 3838 South Warner St., Tacoma, WA 98409), I wrote:

"... there's just so much graphics can do for a game. Two of Leo's recent games, Cubes and Snake Eggs, are games in point. Both are beautifully programmed, with very clever graphics, yet even the most devoted of games players would probably be interested in no more than a few games. After all, marzipan is used to create some beautiful candies, but how much marzipan can you eat?"

I'd like to share Leo's response with you:

"One of the most rewarding things I've

found to do is to program animated graphics games which others like to use for the pleasure they get from them. Most of my programs involve rather trivial games, partly because the game itself interests me less than the graphics, and partly because there is so little memory left in a 16K TRS-80 after the graphics have been set up.

"I tend to look at my programs as little artistic creations which are to be enjoyed for a while and then set aside for a while until the desire to experience them again arises. Much as a person would listen to a special record of music only when the urge comes to do so. But that urge doesn't necessarily occur too often. Your 'marzipan' describes this pretty well.

"Now, I'll have to clear up a misunderstanding. Cubes isn't really a game at all. The ads for it and the instructions with it must be misleading. I wrote Cubes to solve the Instant Insanity puzzle. It is a very special-purpose program! You are supposed to have the real blocks in front of you and you use the computer to find the correct re-orientation for a solution. One could use the program to design a new coding blocks, too, I suppose.

"I wrote that program after spending some hours solving the real puzzle and then forgetting how I did it. The puzzle comes with an offer to send a person the solution if he or she writes to the company for it. I designed Cubes as an alternate method for people to get solution to all such block puzzles."

RNMBR

After you've written a program in which you had to insert lines here and there, you may want to renumber some or all of the lines. Perhaps you want to make the program easier to read, or maybe you prefer evenly spaced line numbers, or you need to insert more lines and there's no place to put them.

You could rewrite the whole program, which might take hours, or you can use a Basic renumbering program, which could take only minutes.

RNMBR is offered by Microbiotic Computing Inc. (1124 Bernalillo Place S.E., Albuquerque, NM 87123) for $20. You order a cassette depending upon your Level II TRS-80 configuration: 4K, 16K without DOS, 32K or 48K with DOS.

The short but complete 14-page manual explains the simple RNMBR method for renumbering. You add one or two of three different types of REM statements to the ends of the lines (or to the first of the group of lines) you want to change.

If you want the renumbering to start with line number 150, just add REM//B=150 at the end of the first line to be renumbered. The default line number, which the program will start with if you don't specify one, is 100.

To create an increment of 7, simply add REM//I=7 where needed. You can put it on the same line as the B command.

If a line number is not to be changed, add REM//S, which means the number stays the same.

RNMBR changes line-number references as necessary in statements such as GOTO, GOSUB, ON/GOTO, IF/THEN/ELSE, etc. It is written in machine language and loads at the top of your memory, which is why you have to specify your memory size. RNMBR is written twice, once on each side of the cassette.

The RNMBR commands can be left in your program, since they're all in REM statements. You might leave them in while developing a program, and remove them when finished.

RNMBR also includes diagnostics in case you make a renumbering error: three fatal-error diagnostics and two informative diagnostics.

The first informative diagnostic flags GOTOs and GOSUBs that aren't followed by a statement number. The other will flag references to line numbers that don't exist.

The first fatal condition occurs if two lines are given the same line number in the renumbered program. The second occurs when a line number or increment goes beyond 65535. The third occurs when a command format error is detected; this is when // is detected and the program does not find an expected valid RNMBR command.

If RNMBR detects a fatal condition, the original Basic program is restored to memory and control is returned to Basic. Without this restoration feature, you could lose one or more lines from your program.

As an example of using RNMBR, take this simple program which is

```
1 REM//B=10 A RENUMBERING EXAMPLE
10 INPUT "RADIUS OF SPHERE";R
14 IF R<0 THEN 10 ELSE IF R>100
   THEN 200
100 V=4/3*3.14159*R*R*R:REM//S
110 PRINT "VOLUME OF SPHERE";V
125 A=4*3.14159*R*R:REM//B=50//I=1
130 PRINT "AREA OF SPHERE";A
175 GOTO 10:REM//B=150
200 END
```

to have the first three lines renumbered from 1-10-14 to 10-20-30. Since no increment is specified, RNMBR uses the default increment of 10. Line 100 stays with the same number, as will 110. But 125 is changed to 50, and line 130 to 51, because an increment of 1 is specified. Line 175 is changed to 150, and because the increment hasn't been changed, line 200 becomes 151.

```
10 REM//B=10 A RENUMBERING EXAMPLE
20 INPUT "RADIUS OF SPHERE";R
30 IF R<0 THEN 20 ELSE IF R>100
   THEN 151
50 A=4*3.14159*R*R:REM//B=50//I=1
51 PRINT "AREA OF SPHERE";A
100 V=4/3*3.14159*R*R*R:REM//S
110 PRINT "VOLUME OF SPHERE";V
150 GOTO 20:REM//B=150
151 END
```

Note that RNMBR automatically changes all line-number references, and moves new lines 50 and 51 to new positions in the program.

RENUM

How does RNMBR compare with Radio Shack's RENUM line-renumbering program, which costs $9.95?

The old TRS-80 catalog said RENUM is for 4K only, and the new one says it's for 16K only. Both are wrong. The package consists of two cassettes, with four RENUM programs (each written twice), for 4K, 16K, 32K and 48K systems. A line-renumbering program is included in Disk Basic.

To call RENUM, you enter SYSTEM, and then a slash followed by the entry-point; for a 16K Level II system, this is /31820. You use this calling sequence every time you want to renumber.

When you call RENUM, the program displays three prompts, and all you do is enter the parameters, one after each prompt. The first is OL#= for the old line, then NL#= for the new line, and INC= for the increment. Default values for the three are 1, 10 and 10, respectively. So if you don't enter a number after OL#=, the entire program will be renumbered.

RENUM has two error messages. The first indicates an illegal function call, if you try to create line numbers greater than 65529, enter an INC of zero, use a value of OL# greater than the largest line number in your program, or try to change the sequence of the program lines. Note that you *can* change the sequence with RNMBR.

The other error message occurs when your original program references a line that doesn't exist, such as GOTO 100 when no line 100 exists in the original program.

RENUM also changes the line-number references in you program, for references in lines with GOTO, GOSUB, THEN, ELSE, etc.

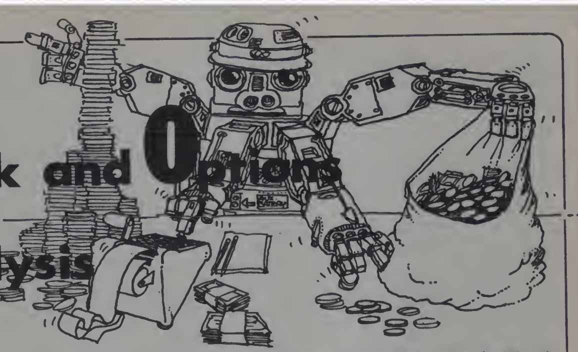
So RENUM costs half as much as RNMBR, but does only half as much. With RENUM, you can't change the sequence of lines, or change only one line number (unless it's the last line in the program).

Using RENUM, you could renumber most of the lines in the sample program that RNMBR does a complete job on. But you'd have to first change line 1 to 10, with an increment of 10. Then change new line 40 to 50 with an increment of 1. Then change new line 52 to 100, with an increment of 10. Then change new line 120 to 150, with an increment of I. But there's no way in RENUM to move those two area lines further up in the program, as RNMBR does.

80 Software Directory

80 Software Directory is the new name of what previously was *TRS-80 Software*

Stock and Options Analysis



Keep the data you need to make timely investment decisions at your fingertips with this incredibly powerful investment tool. Considerable effort has gone into methods of tilting the odds in the investment game. Out of this has come the discovery that the strategy of hedging listed options against common stocks can tilt the odds drastically. In fact, it can be *more conservative* and *more consistently profitable* than the simple buying and selling of stock.

The four programs in this package are designed to be used in the real world, and include the effects of commissions, margin interest and dividends, where applicable. Possible investment attitudes, the listed option markets, puts and calls and option strategies are covered in extensive documentation.

The program **Option** presents important indices of both opening and closing call option transactions. The manual includes sample runs illustrating combination strategies with covered and uncovered calls, and covered and uncovered straddles recieve detailed treatment.

The **Opgraph** program presents a graph or a table, as the user chooses, of profit from any combination of six basic positions: long or short a stock, long or short a call and long or short a put. Sample runs are presented which cover hedging with calls, out-of-the-money hedges and in-the-money hedges.

Newprem enables the user to predict the future premiums of an option at whatever time and future stock price the user selects. This method requires the establishment of a data base of historical option premiums in whatever detail the user desires.

Finally, **Portval** enables the user to determine on an item by item basis, the cost, current value per share, total current value and capital gain of a portfolio consisting of long and short stock, and long and short option positions. This program assists the user in keeping a readily available and easily updatable record of his portfolio and, at the same time, assists him in measuring his progress towards financial success.

In order for an investor to continually improve his performance it is necessary for him to refer to past performance; this requires useful records. Finally, he should constantly be evaluating his performances to assure himself he is playing the right game.

The **Stock and Options Trading Analysis** package is available for the 16K TRS-80 Level II on cassette (CS-3306) and disk (CS-3801) for $99.95. Creative Computing Software should be available at your local computer store. If your favorite retailer does not stock the software you need, have him call our retail marketing department at the number below. Or you can order directly from Creative Computing Software, Dept AGII; P.O. Box 789-M, Morristown, NJ 07960. Visa, MasterCard, or American Express are also welcome. For faster service, call in your bank card order toll free to 800-631-8112. In NJ call 201-540-0445.

creative computing software

TRS-80 Professional Software

TRS-80, continued...

Source (*Creative*, April 1980, p 134). It's published by Computermat (Box 1664, Lake Havasu, AZ 86403).

The format is the same: computerized listings of TRS-80 software, in eight categories, each alphabetized by title and also by supplier.

The undated (summer?) fifth edition is $6 and contains 7,500 listings and 614 suppliers. The price went up to $8 last September.

According to Dave Ahl, who took the time to check out how *Creative Computing* fared, the directory is "singularly poor." He went on to say, "Of the stuff listed in our year-old software catalog, they list only eight packages out of 30. One listing is completely wrong and two others are incomplete."

Dave's advice is "Don't waste your money."

Another problem with the directory, as noted last April, is that you have no way of knowing which, for example, of the 22 versions of biorhythm (10 Level-I, 12 Level-II) are good or bad, and which are copies of some other.

In 1979 the publisher intended to include "software reviews," but so far these haven't materialized.

The directory may be of interest to someone who buys a lot of TRS-80 software, and who realizes the directory is incomplete.

Astrology

After you've loaded the machine-language Astrology program from Radio Shack ($29.95 for 16K Level-I or Level-II TRS-80), it asks for your full name, birthdate, time of birth (AM or PM, daylight savings time?), where you were born (city, state, latitude, longitude), and whether or not you want your horoscope printed.

Among the optional data are the person's name, exact time of birth (a noon birth is the default time), and place of birth.

The 36-page manual includes latitudes and longitudes for 50 major U.S. cities. You can approximate these "using any atlas or a large map," says the manual, which also recommends buying two books, *Longitudes and Latitudes in the United States* and *Time Changes in the USA*, "if you are going to be doing a number of charts."

If you want the horoscope printed, you get a printout nearly a yard long, containing seven sets of data: a listing of the input data; planets in the signs, such as

```
URANUS   IS  18.12 DEGREES IN LEO
```

natal aspects, such as

```
MERCURY IS SEMISEXTILE VENUS
( 30 DEGREES APART)  -4.58
```

elements and modes: planets in the houses, such as

```
JUPITER IS IN THE  4TH HOUSE
```

signs on the cusps of the houses, such as

```
THE CUSP OF THE  1ST HOUSE IS
25.9 DEGREES IN VIRGO
```

and for the grand finale, The Zodiac At Birth:

```
                          THE ZODIAC AT BIRTH

                                 GEM
                                 3.23
              CAN                 I                       TAU
              8.04                                        0.00
               \                  I                        /
                  \     SUN 29.32 GEM I                  /
                     \            I                    /
     MER 19.03 CAN     \          I                  /
                          \       I                /
 LEO                        \     I              /              ARI
 9.54                         \   I            /                1.11
   \    \                       \ I          /            /   /
        \    \                  10      9  /        /   /
VEN 14.53 LEO  \                 \   NATAL   /   /
MAR 12.21 LEO     \          11   ASTROLOGICAL   8
URA 13.40 LEO       \              HOROSCOPE    /
PLU  1.59 VIR          \              FOR     /
                            ALFRED E. NEWMAN
                        12                      7
ASC  7.30 VIR - - - - -    BORN: JUNE     21, 1959
                              10.09 AM  CLT       - - - - - -PIS  7.30
                              4.36  PM  GMT
                         1                        6
                            CLEVELAND, OHIO
                       /      LON   81.42     \     \
                     /        LAT   41.30       \     \
                  /      2                    5    \     \
                /           /                  \      \     \
             /          /       3       4         \      \     \
          /           /           I                 \         \
     /  N.N  9.29 LIB /           I                  \  MOO 10.57 CAP  \
 LIB                              I                    \                 AQU
 1.11                  /   JUP 23.22 SCO  SAT  4.10 CAP                  9.54
                      / NEP  4.18 SCO I                \
                    /                 I                  \
                  /                   I                    \
                /                     I                      \
              /                       I                        \
            SCO                       I                         CAP
            0.00                     SAG                        8.04
                                     3.23
```

If you don't have a printer, the program gives the same material except for The Zodiac At Birth, in seven displays. (The Zodiac At Birth is a Natal Chart, which encompasses most of the data in the displays, but is too large for the screen.)

The manual has 19 pages on Reading the Horoscope, with sections on Influence of the Planets; Natal Aspects; Signs of the Zodiac; Placing the Planets in the Signs; Blending the Sun, Ascendent and Moon in the Signs; The Houses; and Secondary Progressions.

The manual ends with a list of eight books for Recommended Further Reading.

The most fascinating part of the package is the large (35 by 23 inches), elaborate and colorful poster, which has a dozen charts that help interpret the horoscope. Three are of major interest.

According to the "Planets in Signs" section of the poster, I am a detail analyzer, have a need for harmony with others, pay attention to detail, lack cohesion, etc.

According to the "Planets in the House" section, I have these characteristics: optimistic, flexibility, communicator; family loyalty; expresses well, talker, witty; restless, belief in own ideas, etc.

According to the "Signs on the Cusps of the Houses" section of the poster, I am (or have) sensitive, intuitive awareness; ambitious for luxury; dramatic travel; perfection in home, conversation; passionate, jealous, demanding; need excitement, etc.

Most of the characteristics on the three main charts are flattering, with a few pejorative ones thrown in for balance, as is usual with horoscopes.

The Astrology program is played absolutely straight by Radio Shack. That is, the manual is written from the attitude of a true believer, to the point of saying that if you take an "Aries Ascendent person" and put him together with a "Taurus Sun practical, stable car" ("think of the Sun in a sign as the type of car the person drives"), "he'll drive that car at top speed, recklessly."

The Astrology program is listed, appropriately, in the Radio Shack catalog under "Games," along with Flying Saucer, Pyramid, and SpaceWarp.

Beating Microchess

If you're getting highly frustrated from not being able to beat Microchess evenat the IQ 1 level, you may be interested in a note sent by Ted Fisher of Danville, Illinois. He "just wanted to report a piece of top-secret information," a program to beat Peter Jennings' Microchess 1.5 in five moves:

E2-E4
D1-F3
F1-C4
C4-D5
F3-F7

Microchess makes the same moves at all three levels of play, taking more and more THINKING time to do so if you move up form IQ 1 to IQ 2 to IQ 3, and finally comes up with

CHECKMATE
YOU WIN!

Peter Jennings says several people have sent him programs such as this one, and comments that it's similar to the Fool's Mate; "Microchess falls for it." However, Microchess 2.0, the version used in the PET and Apple II computers, "doesn't fall for it," he said.

As for an improved version of Microchess for the TRS-80 from Personal Software Inc., Jennings says there won't be one in the immediate future, but adds, "I suspect we will be forced into developing it by popular demand. We get so many letters asking 'When are you going to send us the improved version?' They just want to see general improvements, nothing much specific. Any improvements at all will mean going beyond the 4K that Microchess is written in now. The next logical step up, for the TRS-80, is to a 16K Microchess."

Microchess is available at $19.95 in many Radio Shack stores no, and was reviewed in this column in the Feb. 1979 *Creative* (page 102).

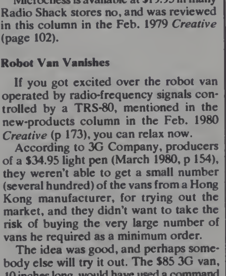
Robot Van Vanishes

If you got excited over the robot van operated by radio-frequency signals controlled by a TRS-80, mentioned in the new-products column in the Feb. 1980 *Creative* (p 173), you can relax now.

According to 3G Company, producers of a $34.95 light pen (March 1980, p 154), they weren't able to get a small number (several hundred) of the vans from a Hong Kong manufacturer, for trying out the market, and they didn't want to take the risk of buying the very large number of vans he required as a minimum order.

The idea was good, and perhaps somebody else will try it out. The $85 3G van, 10 inches long, would have used a command unit plugged into the TRS-80's output port, "operating the van through radio controls," using "simple Basic statements to control forward, reverse, right, left, start, and stop."

It could have been called the "TRS-80 bus".... □



Apple~Cart

Chuck Carpenter

Correspondence is always welcome and a response will be made to those accompanied by a SASE. Send your letters to: Chuck Carpenter, 2228 Montclair Pl., Carrollton, TX 75006.

Well, here we are in another year. Lots of things have happened in the Apple world. And, I expect a lot more will happen. There are a dozen or so companies making accessories and many more writing software. I mentioned a directory of Apple products in last months column. This same company publishes a directory of Apple software. And, I have seen at least one other company offering a directory of Apple software. The magazine is called *Peelings II* and is dedicated to reviewing software for the Apple. Like any review, the contents are biased by the preference and interests of the reviewers. They claim to be impartial but we are, after all, human creatures. Even so, the magazine is a valuable source of information about the software and its general worth. *Peelings II*, at $15.00 for 6 issues, is available from Peelings 11, 945 Brook Circle, Las Cruces, NM 88001; Phone (505)523-5088 evenings. At this time (Sep. '80), the copy 1 have is Vol 1 No. 2. If you're interested, you may want to get all the back issues too.

DOS 3.3

This new Disk Operating System (DOS)from Apple converts your 13 sector system to a 16 sector system. You can still use your 13 sector disks as-is but the process is more awkward. You can't boot directly from 13 sector disks so you have to go through a 2 step process each time.

Included with the DOS 3.3 package are the following:

- Two new ROMS for the controller card
- A ROM puller tool
- The DOS Manual
- A 16 sector Master System diskette
- A Basics diskette

The ROMs are installed in place of 2 that come with the disk drive controller card. The programs in these ROMs allow the system to "read" the 16 sector diskettes. After you install them, you can no longer use 13 sector diskettes to boot the system. To use 13 sector disks, first boot the system with the Basics diskette. Then insert the 13 sector diskette and run. As long as no problems occur you can change from one diskette to another, If you're using the protected type diskettes (not copyable) you have to start over from the beginning to change to another diskette. Same thing if a loss of 13 sector DOS occurs. The 16 sector Basics disk has to be booted first, then run the 13 sector disk.

All is not lost though. Included on the 16 sector Master System diskette are programs to convert (move programs from) 13 sector disks to 16 sector disks. Since you gain about 23K more storage space on a diskette, there is some advantage. Also, the 16 sector system is compatible with the language card system and the new Softcard CP/M and MBasic system from Microsoft. (The ROMs used to convert the controller card are the same as those for the Language Card system.) Once you programs are moved from 13 to 16 sector diskettes, operation is the same as always. You could even initialize the diskettes with new volume numbers and HELLO programs before you transfer programs. (Dave Powell, now stationed in Germany, might appreciate this bit of information.) The transfer of programs is made easy with a program called MUFFIN. This program lets you transfer all types of individual files or complete catalogs with single or multiple drive systems.

Another program included on the System Master diskette is called FID (File Developer). This program has two functions. First, it lets you easily catalog, copy,delete, lock and unlock all types of DOS files. Second, it lets you copy from one diskette to another with only one disk drive. This program extends the capabilities of the system allowing you to more easily work with files and the DOS.

The DOS manual is a revised and expanded version of the original DOS 3.2 manual. Most of the information is the same. There are new sections covering operation using 16 sector diskettes. A section on Format of Diskettes Information is expanded as is the section on Using Machine Language Files,especially the RWTS (Read or Write a Track or Sector) subroutine. Other sections fully describe the features and use of the FID and MUFFIN programs.

1 heard rumblings at one of our Apple Corps meetings that DOS 3.3 would work only on 48K systems. I can't tell you otherwise, since mine is 48K. Check it out first. You can get DOS 3.3 for about $60.00 from your Apple supplier.

Reader Input

Randy Reeves from Cypress, TX sends along a tip for relief of programmers eyes. He has found that the plastic material sold in auto stores for reducing glare in your car, works well for cutting down glare from the monitor. The effect can also be achieved by using the sun screen material sold at hardware stores. Randy is also recommending the Program Line Editor from Synergistic Software. This program is also available from *Call A.P.P.L.E.* if you are a member. The editor is being used by serveral of our club members and is claimed to cut down programming time. I have it but have not

used it enough to make any knowledgeable comment.

In the August'80 column, I commented on the declining quality of disk drives being shipped by Apple. Apparently others have experienced problems similar to mine. From Otterbein College in Westerville, Ohio, David Deeve sent along his techiniques for dealing with the problem. He writes, "I've found it works well to insert the disk, close the door, reopen the door and close it again." (This centers the diskette on the clamp...C.C) And he continues, "Also, when first initializing or copying a disk I always remove it, reinsert it and try a CATALOG to assure me that the disk was properly written." David also included what he finds to be an undocumented Pascal item. "If you have a codefile named SYSTEM.STARTUP (not SYSTEM. STARTUP.CODE) it will run automatically when the disk is booted. That is, a turnkey or "HELLO" type program."

An Applesoft Bug

A letter from Joe Verzulli from Port Jefferson Station, NY, turned up an interesting little Applesoft Quirk. The problem has to do with the use of string variables when DOS is not in use. It was an interesting coincidence that the July/August issue of *Call A.P.P.L.E.* included a short piece describing the same problem. Joe's example program looks like this:

```
10 GET R$
20 T = VAL(R$)
30 PRINT T
```

If you type in a number less than 9 for the GET, you will see the number printed to the 16th power. If you add:

```
15 R$ = R$
```

The problem seems to go away. I got as far as determining that a second variable was added to string variable table. You can do this by examining the area of memory where string variables are stored. In Applesoft, this is from HIMEM down. As you will recall, without DOS, HIMEM is at 49152 or $BFFF (HEX). I puzzled over the reasons for a while and called for help. I called Bob Sander-Cederlof at S-C Software and described the problem. He had just seen the *Call A.P.P.L.E.* item too. After discussing it for a few minutes, I had a better understanding and sent along a reply to Joe. A couple of days later I received a more detailed analysis from Bob to include in the column.

The Details

If you have a 48k Apple, and start up Applesoft without DOS, the program included by Joe and similar ones by others will produce strange results. If the number 1 is entered, the value printed is 1,11111111E16. For 2 you will get 2.22222222E16 and for 9 it becomes 1E17. Continuing essentially as Bob wrote it, this is what happens.

"The VAL function in Applesoft has a bug. I looked into the code (from $E707 through $E745 in the ROMs), and here is what it does. It finds the string, and sets the address of the first byte of the string into $5E and $5F. (Remember that $ means a hexadecimal number...C.C.) Then it adds the length, and stores the address of the byte following the string in $60 and $61. The value stored in that byte is saved on the stack, and a zero put in its place. Then the FIN subroutine is called, to convert the string to a floating point value. After the conversion is finished, the original contents of that byte are restored from the stack.

"The problem is this. In our little program, GET(R$) creates the R$ string at location $BFFF. The byte following is $C000, but there is no memory there. In fact, $C000 is the input register from the Apple keyboard! When VAL tries to store a zero at $C000, nothing happens. The FIN subroutine reads the digit you typed once at $BFFF, and then 16 more times at $C000, $C001,..., $COOF. That is a total of 17 digits, like this: "11111111111111111". The value of such a string is truly 1,11111111E16, just like the Apple said. If you type a 9, the value is rounded up to 1E17.

"Inserting the statement R$=R$ causes Applesoft to create another copy of the string at $BFFE, which is a safe location. Safe, because the zero VAL inserts will go at $BFFF, which is a real memory location.

"Another way to avoid this problem is to use HIMEM:49151 instead of the normal 49152 that is automatically set up. Still another way is to be certain that R$ is not the first use of a string. But my preferred "fix" is to buy a disk drive or two and use DOS. When DOS is loaded (in a 48K machine), HIMEM is at $9600, and everything works fine!"

Bob and I also discussed what would happen in a machine with less than 48K. Our guess is that it will be garbage since there is nothing there. Or, it just might seem to work ok. If you have a less-than-48K machine, try it out. My thanks to Bob for sharing this in-depth analysis.

Micro-Verter

If you haven't bought a modulator for your Apple, try this one. The Micro-Verter by ATV Research is battery operated and requires only one connection to the video output connector on the Apple. No connection is required to the TV since the signal is radiated by a short antenna stub on the Micro-Verter. The modulator is designed to work in the UHF channel range. After I installed the batteries, I experimented with the tuning range. I found that it would tune from channel 14 to 21. The unit seemed to operate the best on channel 17. There was practically zero distortion and very clear color on this channel. My guess is that I was in the center of the tuning range. This would be likely to give the best balance of bandwidth and other desirable characteristics. Speaking of distortion, this unit provides the cleanest signal of all the modulators I have tried. Since you do not have to connect the unit to the TV set, the worms from computer switching harmonics are practically nonexistent. Another feature I found useful also relates to the freedom from connections. If you wanted to use several monitors for demonstration purposes, you can use the Micro-Verter with a 6in hairpin antenna connected to the stub. This way the signal can be used by several sets within a 10 foot or so radius of the modulator. Very handy in a classroom environment. I used it this way when I taught an assembly language class.

The unit is packaged in a metal box and includes all connectors, cables, and a battery holder inside. There is a power switch on the front. Even though the unit will last several months with the switch left on, the switch provides extended battery life. The battteries are not included with the modulator. One note of caution. Use a plastic screw driver to tune the unit to the desired operating channel. This is not mentioned in the instructions. A metal screw driver could distort the tuning and is likely to short the metal case. The Micro-Verter is $35.00 postpaid in the U.S. and Canada, from ATV Reasearch, 13N. Broadway, Dakota City, Nebraska 68731; phone (402) 987-3771.

From Here And There

As a result of collecting Apple information from all over, I have accumulated several clever and useful programming ideas. The first one is shown is listing 1. This is the control program from Southeastern Software's *Magazine Article File Program*. The program sets-up a menu of options. Then, you can use the escape key (ESC) to move the cursor over the selections. Several ASCII values are used in the program. You will see these in the CHR$ functions. These are:

- CHR$(4) (CTRL) D
- CHR$(13) (RETURN)
- CHR$(27) (ESC)
- CHR$(91) left bracket

You can find all the ASCII codes on pages 138 and 139 in the Applesoft reference manual.

Each of the menu options are posi-

```
0 O$ =  CHR$ (13) +  CHR$ (4)
20  HOME
30  PRINT O$;"NOMON C,I,O"
40  VTAB 2: HTAB 9: PRINT "SOUTHE
     ASTERN SOFTWARE"
42  HTAB 7: PRINT "MAGAZINE ARTIC
     LE PROGRAM"
50  PRINT "----------------------
     -----------------"
60  VTAB 8: PRINT "  USE THE "; CHR$
     (91);"ESC] KEY TO MOVE THE C
     URSOR"
70  PRINT "TO YOUR SELECTION AND
     THEN TYPE 'RETURN'"
80  VTAB 12
90  PRINT "  CREATE/ADD FILE"
100  PRINT "  SEARCH FILE"
110  PRINT "  CORRECT FILE"
120  PRINT "  LIST CONTENTS OF EN
     TIRE FILE"
130  PRINT "  TRANSFER FILE AND P
     OINTERS"
140  PRINT "  END PROGRAM"
200 SE = 1:VT = 12
210  VTAB VT: GET SE$
220  IF SE$ =  CHR$ (13) THEN  VTAB
     20: GOTO 260
230  IF SE$ =  CHR$ (27) THEN VT =
     VT + 1:SE = SE + 1
240  IF SE = 7 THEN 200
250  GOTO 210
260  PRINT : VTAB VT: FLASH : PRINT
     "*": NORMAL
265  ON SE GOTO 300,310,320,330,3
     40,350
300  PRINT O$;"RUN CR MAG FILE"
310  PRINT O$;"RUN MAG FILE SRCH"

320  PRINT O$;"RUN CRRT MAG FILE"

330  PRINT D$;"RUN MAG FILE DUMP"

340  PRINT D$;"RUN MAG FILE EXCH"

350  VTAB 23: END
65535  REM
```

Listing 1

tioned on the screen in lines 80 to 140. Lines 200 to 250 move the cursor from selection to selection when the (ESC) key is pressed. The GET command is used to check for the (RETURN) key. When it is pressed, the program branches to line 260 where a flashing asterisk is put next to your selection. Then depending on the selection number, a program is loaded and run. This is also a good way to conserve memory. If all the menu options were too large to fit in memory at once, this technique overlays memory with the current operating program. At the end of the current option, you would run the menu program again.

Here's another one from the Southeastern Software *Newsletter*. In the August '80 issue I included an Integer Basic program called 'puff'. This program would scroll a message across the screen billboard style. For those of you without Integer Basic, here's one in Applesoft. It was written by Dr. Romano and it appeared in *Newsletter* number 7. See Listing 2.

This program, also a billboard scrolling program, is from *The Apple Gram* (Dallas). It's called *Moving Title Demonstration* and was written by Bob Sander-Cederlof. See Listing 3.

This program is a little less obvious so I'll include the description that goes alongwith it. First, the program clears the screen, sets text mode and sets the variable 'Q' equal to the address of the Apple's speaker. Next in line 30, two rows of stars are printed. These are for the title to pass through. In line 80 the title content is defined and made equal to the length of the row of stars; 34 in this example. Moving the title is accomplished in line 90. The title is centered by the limits of the FOR loop, and sound is created by the PEEK(Q) function. Line 95 is a delay loop giving the user time to read the display. In line 110, the program is listed so you don't have to do it.

Another one from *The Apple Gram* and also by Bob, helps you ask questions in the program. Here's how it goes... Many times when you want to ask the user of a program a question, they must answer either "yes" or "no". I must have written a thousand different versions of this kind of question routine. Sometimes I code them in line, and sometimes I am smart enough to write a general subroutine to do it.

```
]LIST
1  HOME
3  VTAB 12
5  INPUT "........ THE BILLBOARD
     SHALL READ ...... ? ";A$
6 A$ = "
                          " + A$: REM *
     ** THE SPACE BETWEEN THE QUO
     TATION MARKS IS 40 SPACES
10  HOME
20 L =  LEN (A$)
30  FOR X = 1 TO L
40 S = L - X
50  VTAB 12
60  PRINT  MID$ (A$,X,40);
70  IF S   40 THEN  PRINT  MID$ (
     A$,1,39 - S)
80  FOR T = 1 TO 150: NEXT T
90  NEXT X: GOTO 30
]
```

Listing 2

Finally, I hit upon a really neat subroutine for answering this kind of question while I was working on a text editor. Here is the code:

```
100 PRINT Q$"(Y/N)";: GET A$:

    IF A$="Y THEN YES=1:RETURN

110 IF A$="N" THEN YES=0:RETURN

120 INVERSE:PRINT"PRESS 'Y' OR 'N'...":

    NORMAL:GOTO 100
```

To use the subroutine, you put the question, without a question mark at the end, into string Q$; then you call the subroutine with a GOSUB. When the subroutine returns, you test the value of the boolean variable YES and take appropriate action. The subroutine handles making sure the user does type either a 'Y' or an 'N', and will not return until this is done. It tells them what to do, and keeps on asking the question until they do it. Here is an example of a calling line:

```
1910 Q$="DO YOU WANT TO SEE THE CATALOG":

    GOSUB 100: IF YES THEN PRINT D$"CATALOG"
```

Notice that the IF statement tests the boolean value of YES, by just "IF YES THEN...". This is because the IF processor in Applesoft (as in Integer Basic) will take the true branch if the value of the expression is non-zero, and the false branch if it is zero.

Microsoft Z80 Softcard

Well, I have the Softcard in my machine now, and so far, I am pleased with it. I have found one problem that I believe to be a bug. You can't open a random file directly from the program. In order to use a random file, it was necessary first to use a sequential file command. If the file did not already exist, the random command would not open one. You can work around it by adding one line before you use the random command:

```
100 OPEN "O",#1, filename:CLOSE
```

Also, I found some typos in the section that explains the use of the file commands. There may be more but I haven't found them yet. Another thing to be aware of is the compatibility with other cards. The Softcard will not work with many of the accesory cards available for the Apple. For instance, the Softcard will not recognize the existence of the D.C Hayes Modem. To use it with the Softcard, special software will be required. If you should want to use this system, you should check to see if you can use other boards you may have.For the most part, I find it has features similar to the Language System. If you are using the Language card and so on, The Softcard will be similar in it's relationship to peripherals.

In addition to the Z80 based circuit board, the system comes with 2 half page manuals and two system diskettes. The manuals are divided between the system description and CP/M in one and Microsoft Basic version 5.0 in the other. The manuals are about half-page size. They

```
]LIST
2  REM
MOVING TITLE DEMONSTRATION
5  REM
WRITTEN BY BOB SANDER-CEDERLOF
JULY 7, 1980
10 Q =  - 16336: TEXT : HOME
30  FOR J = 1 TO 7: VTAB J * 4 +
     2: HTAB 4: FOR I = 1 TO 34:
     PRINT "*";: NEXT I: PRINT :
     NEXT J

80 A$ = "MOVING TITLE DEMONSTRATI
     ON": REM             PUT YOUR
     OWN TITLE HERE
90  FOR I = 1 TO 19 +  LEN (A$) /
     2:P =  PEEK (Q) +  PEEK (Q):
      VTAB 8: HTAB 40 - I: PRINT
      LEFT$ (A$,I)" ": NEXT I
95  FOR I = 1 TO 3000: NEXT I: REM
     DELAY SO PEOPLE CAN READ THE
      TITLE
100  REM PUT THE REST OF YOUR PRO
     GRAM HERE.
110  HOME : LIST ,100: FOR I = 1 TO
     3000: NEXT I: END : REM  TO
     SAVE YOU THE TROUBLE...
]
```

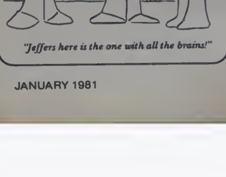
Listing 3

prop nicely in front of you so you can view them easily. There is a lot of information in the manuals so expect to spend a lot of time becoming familiar with the contents. I have never worked with CP/M before, so it was all new to me. I have heard a lot of pros and cons about the merits of CP/M. Once you become familiar with one, no other ever seems quite right. There are a number of utilities to let you do many things that you can't do with the Apple operating system. So learn to use the tools and accept it as another part of the learning experience. Everything is easy once you understand it.

The Basic manual includes a description of the commands and functions of the language. It is intended for the experienced programmer. There are only occasional examples as required to emphasize a point. Also included in the manual are the requirements for calling 6502 routines from the Z80 system. The implementation seems to be well integrated into the two systems since you are not able to detect any interferences. All the keyboard inputs have to pass from the 6502 to the Z80 as do the screen functions and the links to your printer and so on.

Operation of the system is provided by programs included on two diskettes. One diskette is for the standard 13 sector DOS systems. With this disk you get CP/M and MBasic. This version of Basic includes low resolution graphics commands. When this disk is used and MBasic is loaded, you have a little over 14K of memory left. The other diskette is for use with the Language System or with DOS 3.3. The 16 sector system includes hi resolution graphics with Basic as well as the lo res version. In the 16 sector system, you have about 26.5K of memory with MBasic and 17.5K of memory with GBasic.

The system is sure to be well supported by Microsoft. And, eventually there will be Apple CP/M software. I'll be telling you more about the Softcard from time to time.



"Jeffers here is the one with all the brains!"

George Blank

View from the Outpost

The new year brings a lot of promise to Atari owners. We should soon be getting Pascal and a Microsoft Basic, accounting software, and for those of us who prefer to enjoy our computers, a batch of new games. Atari has announced Space Invaders for the 400 and 800 as their first machine language cassette. This game has become one of the most popular arcade and computer games, and I look forward to getting a chance to play it.

We can also expect to see an explosion in Atari software available from sources other than the manufacturer. Just this past October Sensational Software announced Trivia Unlimited, Outdoor Games, Haunted House, Air Traffic Controller, Hail to the Chief, Ecology Simulations I and II, Social and Economic Simulation, Story Time, Oregon Trail, and Stock and Options Trading Analysis. We can only benefit from the wide variety of programs coming out.

There has been a change in the climate for outside software houses in the past year. When appliance type small computers became available, the manufacturers wanted to keep the after market to themselves. In June of 1980 Radio Shack compiled and released through their stores a list of over 1000 programs available from other sources. Exidy and Texas Instuments have been running contests to stimulate software development, with T.I. even sending representatives to most of the major software houses, offering them computers and support in developing software for the 99/4. Apple has cooperated in many ways with software publishers, even allowing mailings to the Apple warranty list.

Atari has been very cooperative from the start. As soon as they knew of outside suppliers they started sending out mimeographed sheets listing programs, prices, and publishers. Recently they invited the sources they knew to submit leaflets for free distribution to the Atari warranty mailing list, and by the time this column comes out many of you should have received the resulting package. If you did not send in your warranty card, you are missing out!

George Blank, Foster Rd., Milford, NH 03055

Bytes, Nibbles, and Bits

The remainder of this column will be discussing the internal workings of the Atari computer. In order to understand what is happening there, you will need to know a little about the binary number system. If you have a binary brain and already think in ones and zeroes, just skip to the next heading while I attempt to explain the binary system to everyone else.

We human beings tend to think numerically in the decimal system, probably because our ancestors used their fingers as counting tools, and they just happened to have ten fingers. Computers have neither fingers nor our intellectual capacity, so they have to rely on a simpler system. While we have ten different numerals, from 0 to 9, for use in our calculations, the computer can only choose between two states,off or on,for which we use the digits 0 and 1.

To a computer, any given operation usually involves a switch or light that is either on or off, an electrical charge that is either positive or negative, or a magnetic charge that is polarized either "north" or "south".

Let us think about a single digit counter on an assembly line.. If it is a decimal counter, it starts at 0, and when the first object passes, it becomes 1, then 2, and so on until it reaches 9. As the next object passes, the counter goes back to zero. If we wish to continue counting, we need to add another digit, which can count 99 objects, or 100 if you know whether the 00 is the starting position or the ending position.

If the counter is a binary counter, it starts at 0, goes to 1, and then has to reset back to 0 again. If you want to count even as far as 3, you need 2 digits. Where each position in the decimal system represents a power of ten, each position in the binary system represents a power of two. Let us take the number 111 in both systems as an example. In the decimal system 111 represents 10 squared, (100) plus 10 to the first power, (10) plus ten to the zero power, (1). The total is one hundred and eleven.

Now, let us consider the same group of three digits as a binary number. It now becomes 2 squared, (4) plus 2 to the first power, (2) plus 2 to the zero power, (1). The total is seven. Since we read numbers from left to right, here is a table of the meanings of the positions in a number under the decimal and binary systems:

Number	1	1	1	1	1	1
Decimal	100,000	10,000	1,000	100	10	1
Binary	32	16	8	4	2	1

Using this system, we can now count in the binary system, with the numbers 0 to seven displayed as follows:

Zero	0 0 0	Four	1 0 0
One	0 0 1	Five	1 0 1
Two	0 1 0	Six	1 1 0
Three	0 1 1	Seven	1 1 1

Each digit in the binary system is represented by a single electrical charge in the computer. This is an actual circuit in the computer that has a positive or negative charge. This amount of information is called a Binary digIT, usually referred to as one BIT. Since the central processor on most computers handles eight bits at a time, we also need a term to refer to a group of eight bits, which can give us numbers from zero to two hundred and fifty five. The common term for eight bits is a byte, and the internal memory of the Atari is organized in bytes. Thus, if you have 32K of memory,

Atari, continued...

you have 32,767 bytes of memory, or 262,136 bits. The reason that 32K is 32,767 instead of 32,000 is that we are organized on the binary system, and 32,000 is not an even power of two. The number 32,767 is two to the 14th power. There is one other term that is occasionally used, for a group of four bits. Four bits, or half a byte, is known as a nibble.

Player Missile Graphics

Ted Nelson wins my nomination as the most organized editor in computerdom. Way back in September he sent out a list of editorial features for *Creative Computing* from October 1980 to December 1981. The features scheduled for this month are graphics and animation, digital music techniques, and interactive video disc articles. As a computer addict, I have no money to spare for a video disc, and my musical ability leaves much to be desired. That leaves me with graphics and animation for this column!

The Atari computers have many powerful graphics features, but the better techniques require special knowledge of the hardware and tricky programming. I hope that by giving a sample program and telling how it works, I can open the door to the mysteries and get many of you started.

Let me begin by acknowledging a great deal of help from Atari, especially from Lane Winner, who gave me the sample program presented here and explained it, and to others on the staff who provided me with well over 1000 pages of documentation, preliminary manuals, printouts, programs, and even memos explaining the working of the computer. I also need to credit Rich Bouchard, SoftSide's brilliant systems programmer, for converting all my hexadecimal addresses to decimal for your benefit. I must confess that I tend to think in hexadecimal.

```
100 DIM A$(10),B$(100)
200 GRAPHICS 8
210 POKE 559,62
220 POKE 53248,120
230 POKE 704,88
240 I=PEEK(106)-8
250 POKE 54279,I
260 POKE 53277,3
270 POKE 53256, 3
280 J=I*256+1024
300 FOR Y=J+120 TO J+137
310 READ Z
320 POKE Y,Z
330 NEXT Y
400 FOR X=48 TO 221:GOSUB 500:NEXT X
410 GOTO 400
500 POKE 53248, X
510 RETURN 100 DIM A$ (10) ,B$(100)
600 DATA 60,60,60,60,60,60,255,255,
 255,255, 255, 255,60,60,60,60,60,60
```

"That Turkey! I told him Graphics *language, not graphic language."*

How The Program Works

Line 210 specifies the regular playfield in single line format as a background, from six possible options.

```
Codes:
61 narrow field (128 dots)
62 regular field (160 dots)
63 large playfield (goes off screen)
45 double line narrow playfield
46 double line regular playfield
47 double line large playfield
```

Line 220 tells the horizontal position register to put player 0 at mid screen (location 120). The regular playfield ranges from 48 to 221. The horizontal position registers are at the following locations in memory:

Player 0	53248
Player 1	53249
Player 2	53250
Player 3	53251
Missile 0	53252
Missile 1	53253
Missile 2	53254
Missile 3	53255

It is possible to combine the 4 missiles into a fifth player.

Line 230 sets the color of Player 0 and Missile 0 to Pink. You can write directly into the color registers starting at 704, or indirectly through the hardware chip starting at location 53266.

Chart of Color Locations

Player/Missile 0	704	or	53266
Player/Missile 1	705	or	53267
Player/Missile 2	706	or	53268
Player/Missile 3	707	or	53269
Playfield 0	708	or	53270
Playfield 1	709	or	53271
Playfield 2	710	or	53272
Playfield 3	711	or	53273
Background	712	or	53274

Understanding the color system reguires binary arithmetic. The least significant bit of the color byte is not used, and is identified with an x in the diagrams. Bits 1 to 3 carry the luminance, from black (0 0 0 x) to white (1 1 1 x). Bits 4 to 7 contain the color, as follows: (b represents the luminance bits)

Grey	(0 0 0 0 b b b x)
Gold	(0 0 0 1 b b b x)
Orange	(0 0 1 0 b b b x)
Red Orange	(0 0 1 1 b b b x)
Pink	(0 1 0 0 b b b x)
Purple	(0 1 0 1 b b b x)
Purple Blue	(0 1 1 0 b b b x)
Blue	(0 1 1 1 b b b x)
Blue	(1 0 0 0 b b b x)
Light Blue	(1 0 0 1 b b b x)
Turquoise	(1 0 1 0 b b b x)
Green Blue	(1 0 1 1 b b b x)
Green	(1 1 0 0 b b b x)
Yellow Green	(1 1 0 1 b b b x)
Orange Green	(1 1 1 0 b b b x)
Light Orange	(1 1 1 1 b b b x)

Line 240 looks at the pointer to the top of memory. The top of memory is identified by 256 byte "pages". Thus, if this location had the number 100 (decimal), the top of memory would be at byte 25,600 (256 times 100). We subtract 8 from this number, because we want to save 8 pages (2048 bytes) for our graphics at the top of memory.

Line 250 places the address calculated above as the beginning of our graphics area in the Player/Missile Base Address Register which is at location 54279.

Line 260 tells the Graphics control register to enable player missile graphics. Only the three least significant bits of this register are active, as follows:

(x x x x x x x 0)	Player missile graphics not enabled
(x x x x x x x 1)	Player missile graphics enabled
(x x x x x x 0 x)	Direct memory access to player graphics disabled
(x x x x x x 1 x)	Direct memory access to player graphics enabled
(x x x x x 0 x x)	Trigger latches disabled
(x x x x x 1 x x)	Trigger latches enabled

Note that once direct memory access is enabled, it continues until it is disabled again. If you press break and list this program, you will see the player as a moving vertical line on the screen. To return to a normal display, you must type POKE 53277, 0.

Line 270 sets the size register for Player 0 to four times normal size. Each player or missile can be the full height of the screen. Actually, they extend beyond the top and bottom. But players can only be 8 bits wide, and missiles 2 bits wide. There are dedicated microprocessor chips in the Atari computers to control these players and missiles. We change the size with a POKE into the size registers at the following locations.

Player 0 size	53256
Player 1 size	53257
Player 2 size	53258
Player 3 size	53259
Size for all missiles	53260

The following values are allowed. Only the two least significant bits are used for the player registers. The missile register is grouped so that each two bits represent a different missile.

Value	Result
0 or 2	Normal size
1	Twice normal size
3	Four times normal size

Line 280 points to a memory location 1024 bytes (1K) after the beginning of the Player Missile Base address we established in lines 240 and 250 for our binary description of player 0. Remember that the player can extend from beyond the top of the screen to below the bottom of the screen. Actually, 256 bytes of information are reserved for player 0, beginning 1024 bytes past our base address.

We now need a memory map of the Player Missile bit map area. Actually, there are two possible memory configurations, depending upon the value in location 559 (see the explanation for line 210 above). If we have single line graphics, with 61, 62, or 63 in location 559, then the memory is used as follows:

Offset from Base Address	Use
0 - 767	Not used
768 - 1023	Missiles 0 - 3
1024 - 1279	Player 0
1280 - 1535	Player 1
1536 - 1791	Player 2
1792 - 2047	Player 3

If double line graphics are used, with 45, 46, or 47 in location 559, you only need 1K, or 4 pages of memory. (see below for program changes.) Then the offset is as follows:

Offset from Base Address	Use
0 - 363	Not used
384 - 511	Missiles 0 - 3
512 - 639	Player 0
640 - 767	Player 1
768 - 895	Player 2
896 - 1023	Player 3

Note that the missiles are stored in memory as a single player. When you are using missiles and drawing your bit map, Missile 0 is bits 0 and 1 of each byte, Missile 1 is bits 2 and 3, Missile 2 is bits 4 and 5, and Missile 3 is bits 6 and 7.

Line 300 selects the memory location for our bit map. We are only going to draw a character 18 bytes high, at about the middle of the screen, so we will leave most of player 0 in the background color, and place information on our character in locations 120 to 137 of the memory reserved for player 0.

In lines 310 and 330, we read our binary shapes from data into the appropriate locations in player zero's reserved memory. Each number is a "bit map" of a horizontal slice of player 0, eight bits or one byte wide. Thus the number 60 represents the binary pattern 0 0 1 1 1 1 0 0. Each bit that is zero represents one location on the screen that is presented in the background color, while each one represents a memory location presented in the color set for player 0.

I selected a simple cross for demonstration program just to illustrate the principle. Here is a bit map of player zero:

Locations	Decimal	Bit map
0 - 119	0	0 0 0 0 0 0 0 0
120 - 125	60	0 0 1 1 1 1 0 0
126 - 129	255	1 1 1 1 1 1 1 1
130 - 135	60	0 0 1 1 1 1 0 0
136 - 255	0	0 0 0 0 0 0 0 0

To create your own players, just map out the bits and convert them into decimal numbers, then POKE them into the display list.

Lines 400 and 410 contain the horizontal positions for our player, and control its movement across the screen. Location 48 is the left edge of the screen and location 221 is the right edge.

Line 500 POKEs the desired horizontal position of player 0 into the position register for player 0. Here is a chart of the memory locations for the horizontal position registers:

Player/Missile	Register
Player 0	53248
Player 1	53249
Player 2	53250
Player 3	53251
Missile 0	53252
Missile 1	53253
Missile 2	53254
Missile 3	53255

Line 600 contains data for the shape of our player, as described above in the description of lines 300 and 310.

If you are going to experiment with this program, I suggest that you add the following line:

```
340 PRINT "POKE 53277,0 : GR.0 : LIST"
```

This way, in order to change the program, all you have to do is press BREAK, move the cursor on top of the statement on the bottom of the screen, and press ENTER to turn off the direct memory access and list your program.

Use these changes to try double line graphics:

```
210 POKE 559,46
240 I=PEEK(106)-4
280 J=I*256+512
300 FOR Y=J+60 TO J+77
```

These changes specify a double line regular playfield, reserve only one page of memory instead of two, and adjust the bit map for the more compact memory storage.

Have fun playing with this information, and share your discoveries with me. The quickest way to reach me is at my home address listed at the bottom of the first page of each month's column. Letters sent through *Creative Computing*, The Boston Computer Society, *SoftSide*, Ramware, or The Software Exchange take extra handling and extra time. I regret that it is not possible for me to talk to you on the telephone or answer correspondence except through the column. □

Edmond H. Weiss, Ph.D.

Too Few Words

Edmond H. Weiss teaches effective writing seminars for business, industry, and government. To contact him, call (609 795-5580.

The subject of this month's column is counterproductive language compression excesses. That is, reader-inhibiting, space-saving text economies. The column aim is a language interpolation principles demonstration for effectuating communication accessibility advantages.

Now, if you found that first paragraph hard to follow, congratulate yourself. You have the good sense not to waste your time in deciphering that cryptic passage.

This column is for people who write paragraphs like my first one: people who use too few words. Although wordiness (too *many* words) is the more common problem, compression (too *few* words) is often harder to recognize and correct.

Compression is the evil effect of a noble cause. Sadly, too many writers—motivated properly to keep their texts short and concise—throw away words that are useful or necessary for understanding. Then, they squeeze the remaining words into a dense, impenetrable mash of sentences, which, though grammatically correct, are barely readable.

I blame most of this compression on three mistakes:

- Careless Stringing
- Reckless Chopping
- Heartless Stripping

CARELESS STRINGING

Among the easiest ways to remove little words from your sentences are, first, stringing modifiers before a noun, second, stringing nouns, and third, stringing modifiers *and* nouns. Easy, but dangerous.

For example, a misguided writer will convert *methods that reduce cost and prevent delay* to *cost-reducing, delay-preventing methods* (a string of modifiers).

Or, the same writer might convert—

This language is easily learned because it permits you to generate reports with statements that resemble plain English.

to—

This language has an easily-learned English-like report generating capability.

True, the second sentence is shorter than the first. But it is also impossible to understand (unless you have read the earlier, longer version).

Even more characteristic of the computer literature is the noun string (excuse me, string of nouns). Obviously, no mere mortal can make sense of such phrases as *language interpolation principles demonstration* (four nouns). But what may be less obvious is that most readers cannot make sense of a string of *two* nouns. What, for example is a *management option*? Is it a choice for managers to make, or one of several ways to manage, or something else? What is a *user evaluation*? Is it an evaluation *of* the users, *by* the users, or *for* the users?

When the modifiers and nouns are strung together, the results vary from tortuous to ludicrous. What can anyone make of—

- non-fossil fuel energy source
- operator-induced failure rate increase problem
- integrated distributed processing plan report
- long range security leak reducing plan
- operational planning materials format design criteria
- structured analysis graphics approach

Some of the strings are just funny. One government agency advertises for a *short contracts expert*. (Tall ones need not apply.) A utility company hires a *buried cable engineer*. Somewhere, I an sure, there is someone who calls himself a *floppy disk salesman*. (he works beside the *fat processing technician*.)

The point is simple. Strings of nouns and modifiers are obscure and ambiguous; their meanings twist and turn as you look at them. Do not, for the sake of economy, drop the little prepositions and function words that would turn *dissolving pulp customer* into *customer for dissolving pulp*. Use your judgment. *Data base* is safe; *data base manager* is safe; even *data base mangement system* may be safe (in some places). But almost no one knows what a *card holder file* might be. And nary a soul can make sense of *contiguous sector reference designator*.

RECKLESS CHOPPING

Much compression is due to overzealous cutting and chopping. Some writers act as though they were preparing telegrams and, therefore, excise all the articles. The "telegrapher" writes:

- Please read attached instructions.
- When creating new file, ensure new name not assigned to old file.

Others act as though they might be assessed or penalized for every extra word they use. They write—

We believed the analyst would solve the problem. instead of—

We believed *that* the analyst would solve the problem. (The omitted *that* in the first version invites the reader to stumble on the phrase *believed the analyst*.)

Still other writers—and even a few professional editors—will hack away at transitional words and phrases. They'll chop every *for example* or *in contrast* or *further*. They'll toss out every introductory *obviously* and *certainly* and *unfortunately*. Unfortunately, these words they have cut are just the words that show the reader the logic and flow of the paragraph. An innocent *to illustrate* ties the second sentence to the first. An *on the one hand* warns the reader that the next two sentences are the contrasting halves of one thought.

Edmond H. Weiss, Ph.D., 1612 Crown Point Lane, Cherry Hill, NJ 08003.

HEARTLESS STRIPPING

Some of this overzealous chopping and cutting is *heartless*: a deliberate attempt to strip the moods and feelings, the intensities and the urgencies, from the text. Desiring to be concise and detached, some writers will mistakenly strip away any extra word that is used to emphasize, or characterize, or underscore a statement.

The heartless strippers remove those little, invaluable words of emphasis: *even*, *only*, *especially*, *unusually*....(Granted, some writers abuse these qualifiers. I am concerned here, though, with the ones that almost never use them, the ones who seem afraid of them.)

To make matters worse, most technically trained writers and editors have a neurotic fear of repetition (redundancy, they call it) and, in stripping repeated words and phrases from their drafts, they deprive themselves of the single most effective way to underscore a point or guide a reader through a difficult idea. To illustrate, consider this splendid, redundant sentence written by Peter Drucker:

> Profit and profit alone will provide the capital for tomorrow's jobs: not just for *more* jobs but for *better* jobs.

The typical technical editor would probably ruin this sentence, turning it into—

> Profit will provide the capital for more and better jobs.

Notice, though, that even though the first sentence is longer, it has a virtue that the second lacks: *It cannot be misunderstood*. The tech editor's version may be logically equivalent to Drucker's but Drucker's cannot be misread. Drucker anticipates the reader and clears away every possible misinterpretation with deliberate repetition and emphasis.

The final point, then, about too much compression (too few words) is that it often comes from a mistaken conception of the reader as a perfect, logical processor of information—someone who can extricate every possible fact and relationship from your logically-tight, efficiently-worded manual or report.

Readers, though, are not like that. They cannot infer and impute all your meanings from a dense, terse statement. They cannot divine the grammar in your string of nouns. And even when they can, they usually won't.

So, emphasize, underscore, and repeat. The best writers—even the most concise ones—do it all the time.

Personal Note

I'll be leaving this location for now. If you have enjoyed these last six columns, please look out for *One Hundred Bugs*, soon to be published by Creative Computing Press. □

Software Legal Forum

The comments and opinions of the author are given for educational purposes only and are not meant to be legal advice. Specific legal questions should be referred to your personal attorney.

Harold L. Novick

Robert Gottschalk, whose name appears as Commissioner of Patents in the landmark Supreme Court case of *Gottschalk v. Benson*, the first so-called "computer program" patentability case, has been on a campaign for software protection since he left the Commissioner's position in the early 1970's. He now practices patent law in Chicago and serves as a consultant and expert witness to other patent attorneys. It seems that Mr. Gottschalk's worst fears may soon be realized.

Some quick background information for those of you who just tuned in. The *Benson* case involved the attempted patenting of a newly discovered mathematical algorithm that was used in digital telephone equipment to convert from binary coded decimal numbers to pure binary numbers. The single sentence legal description of the invention, the "claim", was in the form of a "method of converting signals from binary coded decimal form into binary" and the patent application was directed to "the processing of data by program and more particularly to the programmed conversion of numerical information" in a general purpose digital computer.

The decision of the Supreme Court by Justice Douglas characterized the claimed invention as a mathematical formula that "has no substantial practical application except in connection with a digital computer, which means that (if granted) the patent would wholly preempt the mathematical formula and in practical effect would be a patent on the algorithm itself." In one interpretation of the decision, the Supreme Court was merely adhering to its age-old rule that only applied technology can be patented and that "(p)henomena of nature, though just discovered, mental processes, abstract intellectual concepts, are not patentable, as they are the basic tools of scientific and technological work." In a sense, the Supreme Court denied patent protection because the applicant was claiming not the application of a newly discovered mathematical algorithm, but the algorithm itself.

Harold L. Novick, Patent Attorney, Larson, Taylor & Hinds, Arlington, VA 22202.

If that was all Justice Douglas said in the 6-0 decision (three of the nine justices did not take part in the case), then it would have been a decision that perhaps ended the problems. But Justice Douglas continued: "If these programs are to be patentable, considerable problems are raised which only committees of Congress can manage. ...The technological problems tendered in the many briefs before us indicate to us that considered action by the Congress is needed."

It is clear that the court did not appreciate the nature of the beast that it held not to be patentable. But that is not unusual, considering they were dealing with "computer programs." Try and conceptually define a computer program. Then, ask your friends. If you ask nine friends, you will end up, including yours, with eleven different answers. To put it bluntly, computer programs are a weird bird, unlike anything else.

Consider the copyright side of the question. For those readers who have read last month's *Forum*, there will be an appreciation of the difficulties that a Chicago federal district court judge had when he held that a ROM was not a "copy" of a computer program. (This is the so-called CompuChess case). He said that notwithstanding (i.e., "despite" for those readers for whom legalese does not compute) the assumption of both the plaintiff, Data Cash Systems, Inc., and the defendant, JS&A Group, Inc., that it was a "copy." A "copy" is defined by court decisions to be "a tangible object that was a reproduction of the original work." The judge analogized the ROM to a completed building. At law a building is not a "copy" of the architectural plans upon which it is based. An architectural plan is a technical writing which is capable of being copied only by similar technical writings, i.e., by other plans. A building is the result of plans not a "copy" of them. It follows that at common law, (i.e., judicial interpretation as opposed to legislative statute) a copy of a computer program is another computer program in its flow chart or source phase because these are comparable technical writings. While the ROM is the *mechanical embodiment* of the source program, it is not a "copy" of it. (Emphasis added, court citations deleted).

The Chicago judge has been soundly criticized for his holding. But it was his misappreciation of the nature of a computer program that led to his errors. (Technically, the judge was correct that under the old law, a ROM is not a "copy" because that law, written in 1909, required that a copy must contain information that is perceivable to the naked eye of the reader. The new law, effective in 1978, did not. The judge, not understanding computer programs, applied the wrong law.)

Another view from the copyright perspective of the uniqueness of computer programs is the basic principle that only an expression of an idea is copyrightable, not the idea itself. One copyright expert has asked, "Is it possible to render protectable the 'expression' of a program without necessarily granting a monopoly in its 'idea', i.e., in the methodology or process adopted by the programmer?" 1 Nimmer on Copyright S2.18(J) (1980). For example, a computer program is usually used by reading it into the working memory of a computer. Does this make a copy? or, is this the use of a methodology or process?

John Hersey, a famous author and a Commissioner on the Congressionally established National Commission on New Technological Uses of Copyrighted Works (CONTU), dissented from the commission's recommendation that "computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright." Mr. Hersey likens the object code to a cam. The program, now in its "mature and usable form, is a machine-control element, a mechanical device ..."

"We take it as a basic principle that copyright should subsist in any original work of authorship that is fixed in any way (including books, records, film, piano rolls, videotapes, etc.) which communicate the work's means of expression." But a program, once it enters a computer and is activated, does not communicate information of its own, intelligible to a human being. It utters work. ...The mature program is purely and simply a mechanical substitute for human labor." Commissioner Hersey concludes that "a computer program in the form in which it is capable of being used to control computer operations" should not have copyright protection.

The simple answer to Commissioner Hersey is that he is talking about the use of a computer program. Object code is understandable by humans and can communicate ideas to humans. Many programmers can write in object code. This author wrote many programs in 8008 object code.

Even lawyers arguing before the Supreme Court have trouble agreeing as to what a computer program is. In the oral arguments of two cases about the patentability of "computer programs" heard by the Supreme Court on October 14, 1980, the two lawyers arguing that their respective inventions were patentable and the government lawyer opposing their contention all agreed that computer programs should not be patentable. Where did they differ? The government lawyer, Assistant Solicitor Wallace, claimed that the *Bradley* invention to a computer system (claimed hardware) that included a novel firmware module and the *Diehr* invention to a process of vulcanizing rubber using a programmed computer to control the processing time were computer programs. The Bradley and Diehr attorneys strongly argued that the inventions were not claimed as computer programs.

The CompuChess judge, John Hersey, and Mr. Nimmer, all suffer from the problem of defining a computer program. What is a computer program? Trying to protect it with any present day legal mechanisms such as patents, copyrights, or trade secrets is like trying to ram home a square peg in any one of three round holes.

Robert Gottschalk has been arguing, preaching, and cajoling for almost a decade that computer programs need a new form of protection. They are too valuable to be left alone. The Supreme Court in 1972 told Congress to do something. Nothing has been done. In a sense, computer programs are like "poor old Charlie", left riding on the MTA (the Boston subway), in the Kingston Trio's song, when the fare was increased and he was without funds to get off. They are caught in a legal squeeze.

What should the reader do? What can he or she (or it, for mechanical readers) do? Write your state and federal representative. Get Charlie program off the mechanical and Transfixed Appellation. □

puzzles & problems

The Monkey's Uncle Puzzle

Willard Wordsworth, "The Word Professor" from Camelot University, is back to kick off this session of "Merlin's Puzzler's." Willard has one of his famous "change-the-word" puzzles for us to wrack our brains over. In this type of puzzle the object is to change the word in the top row to the word in the bottom row in the fewest possible moves. During each move the puzzler must change one letter in the word so that a new word is formed. (For example: You can change the word WARM into the word COLD using the following four moves — WARM, WARD, WORD, CORD, COLD). (This puzzle is from "Merlin's Puzzler 3").

A	P	E
M	A	N

A Barrel Of Fun

The donnish looking chap pictured here is cogitating over a legal problem presented to him by one of his students. It seems that a wine merchant had died and left an estate consisting of 21 wine barrels. Seven of the barrels were full, seven were half-full, and, the remaining seven were, alas, empty. The merchant had stipulated that each son was to get an equal share of full, half-full, and empty wine barrels. The problem is to find the simplest method in which to accomplish the terms of the will. The don's first solution was for the sons to drink up all of the wine and then divide up the empty barrels, a solution that while being practical, is not the one that the student had in mind.

Just One Over

A man, being asked how many coins he had in his pocket, replied, "If I divide them by 2, by 3, by 4, by 5, or by 6, I shall always have one over." What number had he?

The Davenport Puzzle

Here's one of Merlin's favorite puzzles.

At the right we have a puzzleboard that has been divided into 36 squares. The puzzler is required to place 12 small coins on this board in such a manner that there will be two coins in each horizontal row, two coins in each vertical row, and, two coins in each of the two corner diagonals. Only one coin can be placed in any one square. (This puzzle appears in "Merlin's Puzzler 1").

If you have a puzzle to share with our readers send it in, and, if Merlin uses it he will send you a copy of one of his books.

Until next month, good puzzling.

Your editor,

Charles Barry Townsend

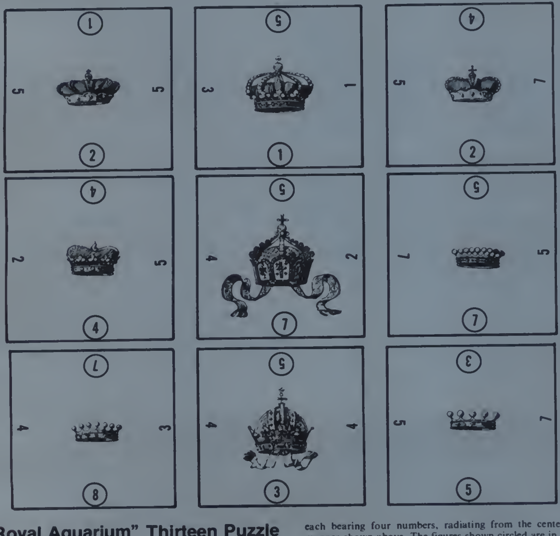
Charles Barry Townsend

The Right Connections Puzzle

This "Brain-Buster" puzzle was contributed by Mr. Wesley M. Shaw of Warner Robins, Georgia. For submitting this problem Merlin is happy to send Mr. Shaw a copy of "Merlin's Puzzler 1".

You are given the task of correctly labeling both ends of a 1 mile long cable assembly spanning the Ochewata River. There are ten identical wires in this cable. You must match corresponding numbers to each end (i.e., #1 labeled wire on one side must match #1 labeled wire on the other side). You are equipped only with 1) 20 one-foot jumper wires, 2) a continuity checker, 3) one round trip ticket on the Ochewata ferry, 4) 10 one-foot lengths of light twine, 5) one bottle of extra strength Tylenol.

Information: 1) The cable is buried beneath the river, therefore, the ends cannot be brought together for direct continuity comparisons (see above illustration). 2) The ends of the wires are stripped but one foot of insulation is exposed before entering the cable. 3) You have all the equipment necessary to solve this puzzle.



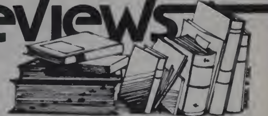
The "Royal Aquarium" Thirteen Puzzle

This is an adaptation of the "Magic Square" idea, but modified in a very ingenious manner, the ordinary processes for forming a magic square being here quite inapplicable.

The puzzle consists of nine cards, not quite 1½ inches each way, each bearing four numbers, radiating from the center, after the manner shown above. The figures shown circled are in the original printed in red. The experimenter is required to arrange the nine cards in a square, the red numbers forming perpendicular lines, and the black numbers horizontal lines, the three figures in each line, whether horizontal or perpendicular, making, when added together, 13. (From "Merlin's Puzzler 2").

Answers on page 192.

Reviews



Computers and Mathematics, by James L. Poirot and David N. Groves. Sterling Swift Publishing Co., Box 188, Manchaca, TX 78652. 463 pages, hardcover $21.95. 1979.

Designed as a text for an introductory course on computers, this book covers a great deal of ground, including flowcharting, Basic and Fortran, error analysis, Boolean algebra, careers, calculators, small computers, statistics, matrices, and 15 pages lifted bodily (with permission from the "Basic Statements and commands" section of Radio Shack's TRS-80 Level II manual.

Such a wide variety is unique in this kind of book, which also includes a dozen pages that list almost 900 references to articles in 14 magazines (including *Creative Computing*), and a four-page bibliography of books about computers, computing, calculators, physics, matrices, etc.

The first appendix contains answers to selected exercises, the second a reference listing of the commands and statements for the H-P 2000, and the third, ditto for the Apple II.

Although this book is laid out in a manner more workmanlike than pretty, it does contain a great deal of information written quite well and in nice detail, so it might be worthwhile considering for classroom use if you teach and intend covering more than just computers.

If you don't teach, this is a wide-reaching introduction to computers and math that you can learn a great deal from, although the section on small computers got out of date in a hurry. Over half the machines listed in a section on Selecting a Microcomputer (taken from an article I wrote for *Popular Electronics* four years ago), are no longer available.

Information Processing, by Marilyn Bohl. Science Research Associates Inc., Chicago, IL. 507 pages, paperback $12.95. Third edition, 1980.

This third edition of a book first published in 1971 by SRA, a subsidiary of IBM, is used by "several hundred colleges and universities," according to the preface.

It is one of the most handsomely produced books on the subject of information processing in recent years. The many well-chosen photographs, flowcharts, drawing and other forms of artwork are all carefully integrated with the text.

The 17 chapters are on An Introduction to Data Processing, An Electronic DP system, Data Representation, Data-Recording Media, I/O Devices, Storage Devices, The CPU, Computer Operations, EDP Systems, Developing a Program, Programming Techniques, Programming Languages, Operating Systems, Files and Data Bases, Advanced Systems Concepts, Data Communications, and Computer Security and Controls.

As the chapter titles indicate, the book covers a wide range of topics. Each chapter ends with discussion topics and references for further reading.

The text, although detailed and well done, is written in a style that is workmanlike rather than conversational, unlike Prof. Gear's book (see review in this issue), which along with Bohl's book would make a nice pair of textbooks for introductory courses on computers.

It can also be recommended to anyone who wishes to learn a great deal about information processing, from WATFIV to flowcharting, from virtual storage to modems.

Steve Gray, et al

Digital Computer Simulation, by Fred J. Maryanski. Hayden Book Co., Inc., Rochelle Park, NJ. 336 pages, hardcover $15.95. 1980.

This textbook was written to serve for a first course in simulation "at the junior, senior, or first-year graduate level in a computer science, industrial engineering, electrical Engineering, or business administration department."

The author assumes "prior high-level language programming and a limited amount of mathematical sophistication," because his book gets into double integrals, chi-square statistics etc.

The emphasis is on supplying the reader with enough background to perform a complete simulation experiment. To this end Maryanski covers specification, design, coding, debugging, analysis, validation and interpretation. Four simulation programming languages—GPSS, Simscript, CSMP, and Dynamo—are presented with detailed examples and exercises.

The basic principles of probability and statistics are included as an aid to the analysis and understanding of simulation results. In the final chapter, the applications and limitations of simulation are discussed to help the reader properly evaluate the impact of an evaluation study.

Each simulation programming language chapter contains one example, which grows as new features of the language are introduced. Sample listings and output from runs are provided with each example. This is one of the very few books where the printer output is reproduced large enough, and dark enough, for every line to be easily read. Except for three printouts in Ch. 3, that is.

More Chess and Computers: The Microcomputer Revolution; The Challenge Match, by David Levy and Monroe Newborn. Computer Science Press Inc., 9125 Fall River Lane, Potomac, MD 20854. 124 pages, paperback $12.95. 1980.

This is a sequel to **Chess and Computers**, by Levy, who also wrote books for Computer Science Press on the 1975 and 1976 U.S. computer-chess championships.

More Chess and Computers is intended to bring the reader of the earlier book up to date on developments that have taken place in the field during the three years previous to its writing in June 1979. Levy says, however, that "it is by no means essential for those interested in the subject to have read the earlier work before they can follow the present one—simply consider **More Chess and Computers** as a state-of-the-art survey."

Levy is famous for betting that no computer program could beat him at chess. The first chapter describes the 1977/8 matches played as part of his challenge, which he won. There is now a $5,000 prize "with no time limit, waiting for someone." Also, Levy is "prepared to wager up to $10,000 that no-one collects the prize before January 1, 1984".

Other chapters are on State of the Art, Blitz Play, Computer Chess Tournaments, and Microcomputers and Chess. In the last chapter, Sargon is described as "currently the best of the microcomputer programs available," and Chess Challenger and Boris as "the most notable" of the systems designed to play only chess.

The appendixes cover computer-chess miscellany, An Unsolved Problem (how do the parameters of a tree search affect the performance of the search, in chess programming), Games From 1977 Tournaments (55 games, unannotated), and a bibliography.

Obviously, this book is strictly for the chess-playing computernik, and as such is very well written and finely detailed.

Problem Solving and Structured Programming in Basic, by Elliot B. Koffman and Frank L. Friedman. Addison-Wesley Publishing Co., Reading, MA 460 pages, paperback $12.95. 1979.

In the preface to this textbook, the authors "have taken a new approach to teaching an introductory programming course in Basic" because they "feel it is important to teach Basic in the same way that other high-level programming languages are taught. If Basic is to be used as a serious tool for software development, then the principles of structured programming must be applied in order to design effective, reliable software that is readily maintained."

To this end the authors use "the top down or stepwise approach to problem solving" and "three pedagogic tools: a data definition table, a flow diagram and a program system chart."

Three versions of Basic are used. "Each new control structure is introduced by first showing its flow diagram pattern and Dartmouth Basic form; afterwards its implementation in (Digital Equipment Corp.'s) Basic-Plus and standard (ANSI) Minimal Basic is described."

"An extensive set of homework programming problems is provided at the end of each chapter, and exercises are inserted in the body of each chapter. Solutions to selected exercises are provided at the end of the text."

The book starts to explain programming on page 11, with a simple payroll calculation that is built up slowly and carefully until by page 17 it is eight lines long.

Then the authors introduce timesharing, and go on to present a 16-line program to "compute trip time and cost," explaining it in full and very nice detail, yet packing much "annotated output" information in only three pages.

The rest of the book continues in the same fashion, offering much detail, relatively short and easily understood programs, plus many well-chosen programming problems exercises.

Once the reader gets used to the troika approach to showing how a particular statement is used in a program, and used to what may be the unique use (in a Basic book) of individual grey-colored memory-cell figures that give both the variable names and the data, he can profit a great deal from the wealth of information and detail presented.

Small Business Programs, by S. Roberts. Elcomp Publishing Inc., 3873L Schaefer Ave., Chino, CA 91710. 119 pages, paperback $14.90. 1980.

These 32 programs are written in Microsoft Basic, and are available on cassette for several personal computers: 7 for PET, 10 for TRS-80, 4 for Sharp MZ80, 11 for Ohio Scientific Superboard.

Many of the 32 programs are small-business programs, such as inventory control, payroll, quotation, invoice, mailing list, depreciation, and sorting accounts. But some are of marginal use to the small-business man: conversion of physics units, hex-to-decimal conversion, alcohol concentration in blood, etc.

Most of the business-type programs are elementary, such as the payroll program, which does not calculate anything other than gross wages and, by subtracting savings and tax amounts found in DATA lines, net pay.

If you're a programmer, you might manage to enlarge some of these programs, and adapt them for your particular needs. If not, $14.90 is a little too much pay for the seven programs that may be of interest to a small-business man: inventory control with economic order quantity, quotation, invoice, inventory, mailing list, depreciation, break-even analysis.

Then again, if you have absolutely no other source for such programs, the price may be right.

Computer Organization and Programming, by C. William Gear. McGraw—Hill Book Co., NY 455 pages, hardcover $22.95. Third edition, 1980.

First published in 1969, this introduction to machine-level programming and computer organization first discusses general principles and programming techniques, and then shows how to implement this information on four processors: the Cyber 170 and Intel 8080 (new to this edition), IBM 370 and DEC PDP-11.

The third edition also features a stronger emphasis on programming style and includes more material on processes, interrupt handling, I/O processing, and multiprogramming/multiprocessing.

Aimed at students who have taken one or two earlier courses in a procedure-oriented language, this edition discusses a specific microprogrammed processor (the AM 2901) instead of a hypothetical machine. The additional material replaces a chapter on compilers, which has been dropped.

The writing style is semi-conversational, the coverage very thorough, and the book's layout a model of clarity. This is probably the best textbook available on the subject. It is one of the three dozen books in McGraw-Hill's Computer Science Series.

The Wounded Land by Stephen R. Donaldson. Ballantine Books, New York. 497 pages, hardbound. $12.95. 1980.

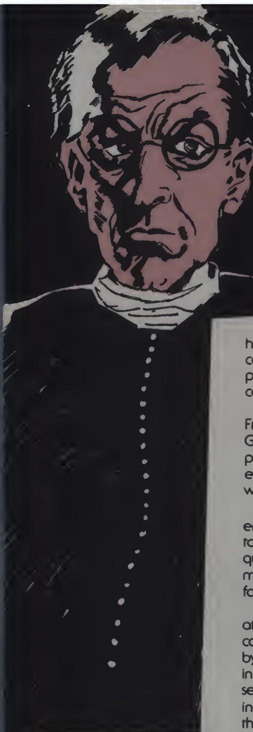
In real life, Thomas Covenant is a leper—that is his most salient characteristic. The ostracism he has suffered as a result of his physical condition has turned him into a thoroughly unlikable protagonist. In Donaldson's first trilogy, **The Chronicles of Thomas Covenant, The Unbeliever**, Covenant is an unwilling participant in a struggle between the forces of good and evil in a strange and beautiful land to which he has been summoned by Lord Foul the Despiser.

By the end of the third book, he has saved the Land, and fantasy fans the world over have heaved a collective sigh of relief, for the Land is a wonderful place and its inhabitants charming and noble folk.

In **The Wounded Land**, Covenant is back—once again summoned by Foul—but this time with a female companion, Dr. Linden Avery, whom he hardly knows. Several thousand years have passed and the Land has changed: its beauty has been destroyed and its people live in fear of the Sunbane which rules their lives. The power and magic which were used for good in the first books have been perverted and the sun turned into an instrument of torture which requires constant infusions of human blood for sustenance. Donaldson does a masterful job conveying the horror of this new evil: for weeks after reading the book I caught myself cringing at the mere mention of the word "sun."

Donaldson is storyteller, but there is more to his books than just a storyline. His characters are complex and his ideas thought-provoking. An added joy is his writing style, which, in a time when such things are accorded little importance, is captivating. His characters have an enormous vocabulary and use constructions which readers may not find familiar. The Land and its residents, however, are subject to few of the same rules that govern us, so their somewhat formal and slightly archaic speech does not seem the least bit out of place.

Donaldson fans who have read the first trilogy will not be surprised to find the same high quality writing in **The Wounded Land**, nor will they find it easy to wait for the publication of **The One Tree**, the next installment in the second trilogy. To those who have never read one of his books, I commend them all, although it is not necessary to have read **The Chronicles of Thomas Convenant** to enjoy **The Wounded Land**.—EBS



Only Fiction . . . or is it?

Tales of the Marvelous Machine:

35 Stories of Computing

A robot friend. A computer God. Artificial intelligence challenging human intelligence in a life and death struggle. A detective solving a computer murder. Computers tricking people or people tricking people with computers. A computer with a soul. Or power. A lonely computer. Or one in love with its operator.

In thirty-five wonderful stories about computers, authors such as Frederick Pohl, Charles Mosmann, M.V. Mathews, Carol Cail, and George Chesbro depict a life in which computers affect the way people live, think, and relate to each other. Interested in what the effect of computer saturation might be? Only fiction can so wonderously dramatize future life.

The book is fun, and will provide wonderful hours of entertainment. For the reader interested in a structured approach to understanding the potential roles of the computer, or wanting quickly to locate stories that support or challenge his viewpoint, a multiple table of contents is provided. This lists the stories in fourteen different categories.

For example, a list of stories in which the computer takes on the attributes of a human separates them from those in which the computer is only an intelligent machine. The stories are categorized by whether they clarify, improve, or worsen the human lot. Stories in which the computers have capabilities available today are separated from those in which the capabilities could be available in the future. There is a listing of the wildly whimsical stories and those in which the computer is utilized in a unique fashion.

Can criminals be caught by computer? Does computer crime pay? Do computers fall in love? Are we all part of a larger organic computer? Here are 35 tantalizing tales that will open your eyes to a new perspective of computers.

Skillfully drawn illustrations augment the stories, giving glimpses of scenes as envisioned by 20 talented artists. This artwork adds another dimension to the text.

Tales of the Marvelous Machines: 35 Stories of Computing, edited by Robert Taylor and Burchenal Green, is a beautiful big 8½" × 11" softbound anthology of 272 pages. 12B

It is available for $7.95 plus $2.00 shipping and handling per order from Creative Computing, P.O. Box 789-M, Morristown, NJ 07960. NJ residents add 5% sales tax. Visa, MasterCard and American Express orders are welcome. For faster service, call in your bank card order toll free to **800-631-8112** (in NJ call 201-540-0445). Or use the handy order form bound into this magazine.

creative computing press

CIRCLE 350 ON READER SERVICE CARD

Creative Computing-- Albert Einstein in black on a red denim-look shirt with red neckband and cuffs.

Creative's own outrageous Bionic Toad in dark blue on a light blue shirt for kids and adults.

Plotter display of Pi to 625 Places in dark brown on a tan shirt.

Give Your Tie a Rest.

I'd rather be playing spacewar-- black with white spaceships and lettering.

T-shirts available in adult sizes S, M, L, XL; and in children's sizes (Bionic Toad, Spacewar and Program Bug) S, M, L. Made in USA. $5.00

Computer Bum-- black design by cartoonist Monte Wolverton on gray denim-look shirt with black neckband and cuffs.

The Program Bug that terrorized Cybernia in Katie and the Computer is back on this beige t-shirt with purple design. You can share the little monster with your favorite kid.

Roll down the block with this little black Robot Rabbit (on a bright orange t-shirt) on your back and you can intimidate every carrot, radish or cuke in your way.

To order, use bound-in card in back of magazine.

index to advertisers

Reader Service	Advertiser	Page
102	Aardvark Technical Service	69
101	Acorn Software	71
103	Adwar Video	121
105	American Square Computers	154
106	Anything Goes Software	154
116	Apple Computer	9
106	Apple Jack	121
109	ASAP	115
116	ASCII	166
110	Automated Simulations	39
111	Banzai Software	42
117	Barclay Bridge	127
*	Basic Magazine	21
112	Basics & Beyond	186
114	Beagle Brothers	187
115	Berliner Computer Center	154
129	Broderbund Software	127
119	C & S Electronics	167
213	CAP Electronics	143
121	California Software Associates	154
123	Chase Manhattan Bank	25
159	CJM Inc.	166
113	CLOAD Magazine	137
133	Color Software	123
130	Compumax	123
134	Compuserve	23
126	Computer Corner of White Plains	154
124	Computer Information Exchange	106
146	Computer Programs Unlimited	166
126	Computer Shopper	169
149	Computer Station	95
*	Computer Systems Design	115
198	Computer Systems International	155
145	Computers "R" Us	117
125	Computerware	115
132	Computers Wholesale	125
137	Computronics	134-135
138	Computronics	37
156	Continental Software	105
162	Cornsoft Group	133
211	Corvus Systems	7
161	Cottage Software	127
171	CPU Shop	63
120	Dakin5 Corp.	45
122	Dakin5 Corp.	117
206	Data Soft	47
131	Discount Software Group	139
195	Disk-O-Tape	154
*	Dresden Associates	101
136	Dynacomp	31
154	Ecosoft	171
139	Educational Courseware	157
164	Edu-ware	75
142	Electronic Specialists	151
150	Exatron	Cover 3
140	Frederick Computer Products	159
*	Galaxy	159

Reader Service	Advertiser	Page
127	Geo-Computers	150
141	Global Parameters	100
135	H & H Trading	151
216	Hayden Book Company	161
153	Hayes Microcomputer Products	Cover 2
147	Heath Company	33
201	Bill Hindorff	150
144	Huntington Computing	169
205	Insiders Software Consultants	107
207	Integral Data Systems	17
208	Interpretive Education	139
209	Iridis	137
219	Krell Software	67
*	Lifeboat Associates	12-13
178	Mastertype	177
151	Dann McCreary	154
221	Med Systems Software	159
196	Methods Research Corp.	159
222	Micro Architect	154
155	Micro Lab	63
156	Micro Lab	62
212	Micro Learningware	154
163	Micro Management Systems	65
166	Microsoft Consumer Products	15
165	Micro Systems Software	137
233	Microtrend International	165
234	Minnesota Software	173
224	Misosys	146
157	Mini Micro Mart	153
152	Monument Computer Service	146
179	Mosaic Electronics	141
191	Mountain Computer	1
225	Muse Software	19
226	Muse Software	113
227	Mytee Music	157
167	NEBS, Inc.	119
160	New Earth Television Works	57
203	Nilonel Mfg.	123
214	NRI Schools	77
172	Ohio Scientific	Cover 4
174	Omni Communications	119
166	Orion Software	129
169	Pacific Exchanges	146
204	Pacific Exchanges	177
239	Peripherals Plus	79,111
240	Personal Computer Systems	5
170	Personal Software	2
228	Professional Data Systems	119
175	Program Store	87
230	Programma International	51
243	Quality Data Supply	157
231	Quality Software	27
173	R-Alpha	121
188	Race1 Computes	141
*	Rainbow Computing	141
176	Rainbow Marketware	173

Reader Service	Advertiser	Page
146	Radio Shack	41
166	Radio Shack Sales Center	155
245	RCA	109
244	RCA	161
*	Retail Roster	152
177	RTR Software	171
247	S-100 Microsystems	55
235	Scholastic Magazine	157
223	Sebree s Computing	157
236	Sirius Systems	157
194	Small Business Applications	34
163	The Software Exchange	96-97
210	Software Technology for Computers	155
189	Spectrum Software	131
167	Sublogic	171
199	Sybex	53
107	Syntauri Limited	104
200	Syntonic Software	179
215	Sync	147
202	Tarbell Electronics	143
246	Taso	173
193	Total Information Service	149
161	Transnet Corp.	118
162	U.S. Robotics	189
184	U.S. Robotics	149
165	U.S. Robotics	173
190	Voicetek	149
192	Xtra-soft	149

Creative Computing

300	Adventure	73
300	Air Traffic Controller	91
300	Atari Software	177
350	Back Issue Sale	143
350	Be a Computer Literate	49
350	Basic Computer Games	190
350	Best of Creative Computing	11
300	Brain & Strategy Games	125
350	Computer for Kids	153
350	Computer Coin Games	183
350	Computer Music Record	61
300	Ecology Simulations	145
300	Investment Analysis	167
350	Katie and the Computer	59
300	Outdoor Games	127
300	Sorcerer Graphic Games	124
350	Sourcebook of Ideas	183
300	Space & Sports Games	175
300	Super Invasion	129
*	Subscriptions	43
350	T-shirts	191
300	Voodoo Castle	161
350	Warehouse Sale	65

*Write advertisers directly

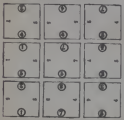
puzzle answers

The Monkey's Uncle Puzzle: APE, ARE, ERE, ERR, EAR, MAR, MAN.

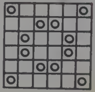
A Barrel of Fun: Pour four of the half-barrels together. We now have nine full barrels, three half-barrels, and nine empty barrels. All can now be evenly divided.

Just One Over: The answer is 61, being the least common multiple of 2, 3, 4, 5, and 6 (60) + 1.

The "Royal Aquarium" Thirteen Puzzle:



The Davenport Puzzle:



The Right Connections Puzzle:

1) On first side of river use jumpers as indicated for identification and label one wire #1 (see Fig. 1).

2) Use half of your round trip ticket.

3) On the other side identify groups and bundle them with twine. Label #1 now, it's the only one not shorted to any other wire (see Fig. 2).

4) Using jumpers, short #1, 2, 4 and 7; short #3, 5 and 8; short #6 and 9.

5) Use the other half of your ticket.

6) Back on the first side of the river remove the jumpers and group them (as they were jumpered) with twine.

7) Now check for the wire which does not show continuity with any other wire. This is labeled #10. #1 is already labeled. Check the twin bundle for continuity with the one labeled #1 and label it #2, and the other as #3. Check the triple bundle for continuity with #1 and label it as #4, check for continuity with #3 and label #5. The other is #6. Go to bundle of four wires and check for continuity with #1 and label it #7. Check for continuity with #3 and label it #8. Check for continuity with #6, and label it #9.

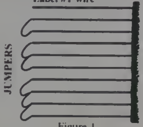


Figure 1.

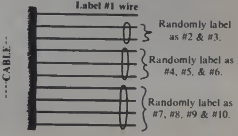


Figure 2.

creative computing

(Please Print)

Name ______

Title ______

Address ______

City ______

State ______ Zip ______

1. How many children live at home between ages 6 and 18?
- A. None
- B. One
- C. Two
- D. Three
- E. Four or more

2. How much money do you spend annually on Software
- A. Less than $50
- B. $50-100
- C. $100-250
- D. $250-500
- E. over $500

3. Your annual income
- A. Under $10,000
- B. $10-15,000
- C. $15-20,000
- D. $20-25,000
- E. $30,000 or over

4. Which type of Printer do you own or operate
- A. Thermal
- B. Dot-Matrix Impact
- C. Type ball printer
- D. Print thimble
- E. Daisy wheel
- F. Electrostatic
- G. At present do not own a printer

Is it—
- A. Friction Feed
- B. Tractor Feed

101	102	103	104	105	251	252	253	254	255
106	107	108	109	110	256	257	258	259	260
111	112	113	114	115	261	262	263	264	265
116	117	118	119	120	266	267	268	269	270
121	122	123	124	125	271	272	273	274	275
126	127	128	129	130	276	277	278	279	280
131	132	133	134	135	281	282	283	284	285
136	137	138	139	140	286	287	288	289	290
141	142	143	144	145	291	292	293	294	295
146	147	148	149	150	296	297	298	299	300
151	152	153	154	155	301	302	303	304	305
156	157	158	159	160	306	307	308	309	310
161	162	163	164	165	311	312	313	314	315
166	167	168	169	170	316	317	318	319	320
171	172	173	174	175	321	322	323	324	325
176	177	178	179	180	326	327	328	329	330
181	182	183	184	185	331	332	333	334	335
186	187	188	189	190	336	337	338	339	340
191	192	193	194	195	341	342	343	344	345
196	197	198	199	200	346	347	348	349	350
201	202	203	204	205	351	352	353	354	355
206	207	208	209	210	356	357	358	359	360
211	212	213	214	215	361	362	363	364	365
216	217	218	219	220	366	367	368	369	370
221	222	223	224	225	371	372	373	374	375
226	227	228	229	230	376	377	378	379	380
231	232	233	234	235	381	382	383	384	385
236	237	238	239	240	386	387	388	389	390
241	242	243	244	245	391	392	393	394	395
246	247	248	249	250	396	397	398	399	400

January 1981 • Expires April 1, 1981

creative computing

(Please Print)

Name ______

Title ______

Address ______

City ______

State ______ Zip ______

1. How many children live at home between ages 6 and 18?
- A. None
- B. One
- C. Two
- D. Three
- E. Four or more

2. How much money do you spend annually on Software
- A. Less than $50
- B. $50-100
- C. $100-250
- D. $250-500
- E. over $500

3. Your annual income
- A. Under $10,000
- B. $10-15,000
- C. $15-20,000
- D. $20-25,000
- E. $30,000 or over

4. Which type of Printer do you own or operate
- A. Thermal
- B. Dot-Matrix Impact
- C. Type ball printer
- D. Print thimble
- E. Daisy wheel
- F. Electrostatic
- G. At present do not own a printer

Is it—
- A. Friction Feed
- B. Tractor Feed

101	102	103	104	105	251	252	253	254	255
106	107	108	109	110	256	257	258	259	260
111	112	113	114	115	261	262	263	264	265
116	117	118	119	120	266	267	268	269	270
121	122	123	124	125	271	272	273	274	275
126	127	128	129	130	276	277	278	279	280
131	132	133	134	135	281	282	283	284	285
136	137	138	139	140	286	287	288	289	290
141	142	143	144	145	291	292	293	294	295
146	147	148	149	150	296	297	298	299	300
151	152	153	154	155	301	302	303	304	305
156	157	158	159	160	306	307	308	309	310
161	162	163	164	165	311	312	313	314	315
166	167	168	169	170	316	317	318	319	320
171	172	173	174	175	321	322	323	324	325
176	177	178	179	180	326	327	328	329	330
181	182	183	184	185	331	332	333	334	335
186	187	188	189	190	336	337	338	339	340
191	192	193	194	195	341	342	343	344	345
196	197	198	199	200	346	347	348	349	350
201	202	203	204	205	351	352	353	354	355
206	207	208	209	210	356	357	358	359	360
211	212	213	214	215	361	362	363	364	365
216	217	218	219	220	366	367	368	369	370
221	222	223	224	225	371	372	373	374	375
226	227	228	229	230	376	377	378	379	380
231	232	233	234	235	381	382	383	384	385
236	237	238	239	240	386	387	388	389	390
241	242	243	244	245	391	392	393	394	395
246	247	248	249	250	396	397	398	399	400

...SCRIPTIONS TO CREATIVE COMPUTING

☐ New
☐ Renewal

Term	USA	Canada and Foreign surface	Foreign air
...s (36 issues)	☐ $53	☐ $80	☐ $143
...s (24 issues)	☐ $37	☐ $55	☐ $97
(12 issues)	☐ $20	☐ $29	☐ $50

...a change of address, please attach old label

...KS, POSTERS, RECORDS, GAMES

...n.	Cat.	Title	List Price	Total
	3G	Binary Dice	$1.25	
	5G	Computer Myth Posters	3.00	
	6A	Best of Creative Computing-Vol. 1	8.95	
	6B	Best of Creative Computing-Vol. 2	8.95	
	6C	Basic Computer Games	7.50	
	6C2	More Basic Computer Games	7.50	
	6C4	More Basic Games-TRS-80	7.95	
	6F	Best of Byte	11.95	
	6G	Colossal Computer Cartoon Book	4.95	
	6H	Be A Computer Literate	3.95	
	6Z	Computer Rage Game	8.95	
	9Y	Problems for Computer Solution Teacher's Edition	9.95	
	9Z	Problems for Computer Solution	4.95	
	10R	Computer Coin Games	3.95	
	12A	Katie and the Computer	6.95	
	12D	Computers in Mathematics-A Sourcebook of Ideas	15.95	
	12E	Impact of Computers on Society and Ethics: Bibliography	17.95	
	12C	Best of Creative Computing-Vol. 3	8.95	
	12B	Tales of the Marvelous Machine	7.95	
	12G	Computers for Kids-Apple	3.95	
	12H	Computers for Kids-TRS-80	3.95	
	12J	Computers for Kids-Atari	3.95	
	CR101	Computer Music Record	8.00	
	RWP	Word Processing Reprint	.50	
	RSS	Sorting & Shuffling Reprint	.50	

SUBTOTAL ______

SOFTWARE T-SHIRTS BACK ISSUES

Quan.	Cat.	Title	Price	Total

Books, Records, Games ______

$2 USA, $4 Foreign Shipping Charge ______

NJ residents add 5% sales tax ______

Magazine subscriptions ______

TOTAL ______

☐ Check or money order enclosed (U.S. funds only)

☐ VISA ☐ Master Charge ☐ American Express

______
Card number

______ ______
Expiration Date Signature

Name ______

Address ______

______

City ______ State ______ Zip ______

creative computing
P.O. Box 2976
Clinton, Iowa 52735

Place
Stamp
Here

creative computing
P.O. Box 2976
Clinton, Iowa 52735

Place
Stamp
Here

Place
Stamp
Here

creative computing
P.O. Box 789-M
Morristown, N.J. 07960

Educator, Entertainer, Accountant.

Your Challenger Personal Computer.

Through the miracle of modern technology, a complete computer as powerful as the multimillion dollar room-sized computers of a few years ago can be put in a package the size of a typewriter and sells for as little as a color television set!

Through its years of microcomputer experience, Ohio Scientific has effectively channeled this tremendous computer power into a "friendly" computer with hundreds of personal uses, via a huge software library of programs for a broad range of personal, home, educational and business use.

This available software allows you to use and enjoy your computer without becoming an expert. The Challenger, however, is a powerful, general purpose computer which can be programmed in several languages by those who choose to.

Here are just a few of the popular uses of an Ohio Scientific Challenger Computer:

Education

The personal computer is the ultimate educational aid because it can entertain while it educates. Software available ranges from enhancing your children's basic math, reading and spelling ability, through tutoring high school and college subjects, to teaching the fundamentals of computers and computer programming.

Entertainment

Many of the Challenger's games educate while they entertain, from cartoons for preschoolers to games which sharpen mathematical and logical abilities. But, entertainment doesn't stop here. The Challenger's graphics capabilities and fast operation allow it to display action games with much more detail than the best video games, providing spectacular action in games such as Invaders, Space Wars, Tiger Tank and more! All popular sports such as golf, baseball and bowling are available as simulated computer games as well as many conventional games such as chess where the computer plays the role of a formidable opponent.

Accounting

Your Challenger computer can keep track of your checkbook, savings account, loans, expenses, monitor your calorie intake and your biorythms.

If you are involved in a business, you can use it to do word processing; accounting, inventory control, order processing, customer lists, client records, mailing labels and planning.

And more:

This may seem like a lot of uses, but it's only the tip of the iceberg for a general purpose computer. For example, your Challenger can be expanded to control lights and appliances, manage your energy usage and monitor for fire and break-ins. Furthermore, it can communicate with you, with other computers and the new personal computer information services over the telephone.

In fact, the uses of general purpose, personalized computers are expanding daily as more and more people discover the tremendous capabilities of these new technological wonders.

Ohio Scientific offers you four personalized computer systems starting at just $479.

For a free catalog and the name of the dealer nearest you, call 1-800-321-6850 toll free.

OHIO SCIENTIFIC

1333 SOUTH CHILLICOTHE ROA
AURORA, OH 44202 • (216) 831-5600

CIRCLE 172 ON READER SERVICE CARD