November 1983 $3.95 U.K. £2.55

Just for your TRS-80® COLOR TDP System 100 Dragon & MC-10

The RAINBOW®

THE COLOR COMPUTER MONTHLY MAGAZINE

Two Great Simulations . . .

Announcing the Winners of Our Contest

DATA COMMUNICATIONS

RAINBOARD SET UP YOUR OWN BULLETIN BOARD!

COMPOSE OFFLINE MESSAGES

BULLETIN BOARDS ACROSS AMERICA

BECOME A 'PRINTER ARTIST'

STORE A YEAR'S APPOINTMENTS

PLUS . . .

AN AUTOMATIC PHONE DIALER

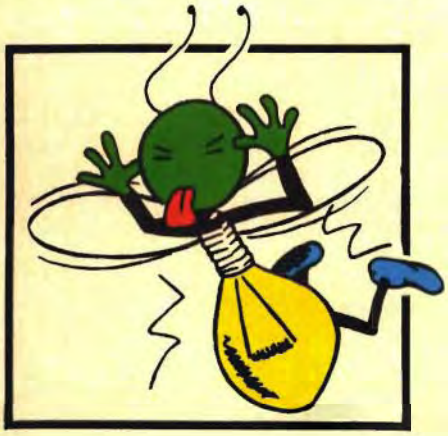
. . . AND MORE THAN 30 PRODUCT REVIEWS

0 11 44254 00001

Under the Rainbow

Feature Articles

74



220

308

COVER art © 1983 by Fred Crawford

Departments

Don Inman's *Using Graphics* will return in December.

Product Reviews

NEXT MONTH: December is our holiday issue month, with a special sackful of stocking stuffers that includes Christmas graphics, Christmas carols, cards and more.

We will also be featuring articles and programs by some of our best-known and most-called-for contributors, including Jorge Mir, Roger Schrag, Larry Konecky and others, as well as our regular cluster of informative tutorials by *Rainbow's* contributing editors.

Plus . . . well over 300 pages of articles, program listings, software and hardware reviews and advertising of products just for CoCo—more information on your Color Computer than is available anywhere else. Happy Holidays!

The Rainbow

November 1983 Vol. III No. 4

Editor and Publisher
Lawrence C. Falk

Managing Editor James E. Reed

Senior Editor Courtney Noe

Technical Editor Dan Downard

Copy Editor Susan Remini

Submissions Editor Jutta Kapfhammer

Editorial Assistants Valarie Edwards, Wendy Falk, Suzanne Kurowsky, Noreen Morrison, Joyce Palgy

Contributing Editors Bob Albrecht, Steve Blyn, Tony DiStefano, Don Inman, Joseph Kolar, Dennis Lewandowski, Bill Nolan, Dale Peterson, Michael Plog, Ph.D., Charles Roslund, Dick White

Art Director Sally Nichols
Assistant Art Director Jerry McKiernan
Designers Peggy Henry, Neal C. Lauron

Advertising Manager Charlotte Ford
Advertising Assistant Lynda Wilson
(502) 228-4492

General Manager Patricia H. Hirsch

Assistant General Manager for Finance
Donna Shuck
Billing Clerk Monica Wheat

Administrative Assistant to the Publisher
Marianne Booth

Customer Service Manager Ivanka Kleier

Research Assistants Laurie Falk, Deidra Henry, Tanya Holder, Wanda Perry
Dispatch Mark Herndon

The Rainbow is published every month of the year by FALSOFT, INC., 9529 U.S. Highway 42, P.O. Box 209, Prospect, KY, 40059. Phone (502) 228-4492. **The RAINBOW** and the Rainbow logotypes are ® Trademarks of FALSOFT, Inc.

Second class postage pending at Louisville, KY. POSTMASTER: Send address changes to *Rainbow*, P.O. Box 209, Prospect, KY 40059. Fowarding Postage Guaranteed.

Subscriptions to **the RAINBOW** are $22 per year in the United States. Canadian and Mexican rates are U.S. $29. Surface mail to other countries is U.S. $57, air mail U.S. $85. All subscriptions begin with next available issue.

Limited back issues are available. Please see notice for issues which are in print and costs. Payment accepted by VISA, MasterCard, American Express, Cash, Check or Money Order in U.S. currency only.

letters to RAINBOW

RO-COCO TO GO

WHICH WAY TO SPEED?

Editor:

In your September 1983 issue on page 234 you stated that *POKE 65495,0* can speed things up but the operator cannot use I/O operations (*CLOAD, CSAVE,* etc.).

However, on page 6 in the same issue you printed a letter from Anthony Graves, with good instructions for using various *POKEs* to accomplish the I/O operations while in the high speed mode.

Who is correct?

Max M. James
Lebanon, IL

Editor's Note: *POKE 65495,0* can be used to speed up the CPU before tape I/O but the tapes will not be at a standard Baud rate.

IT'S TERMINAL

Editor:

Is there any way to hook a Color Computer directly to a R.S. Model III? (For things like common programs — loading and downloading, etc.?)

Mark Rogstad
Yakima, WA

Editor's Note: One method of communication is by use of a terminal program in both computers using the RS-232 ports. Another method is by using CIII by Computer Shack to directly transfer disk files.

HINTS 'N' TIPS

Editor:

In the September issue of *the Rainbow,* you published a program entitled *Bibliography File.* I typed in the program, but was unable to get the title search routine to print to the screen or my printer. I added the following line, and the routine functioned perfectly:

```
3445 IFLEN(TE$)<32 THENTE$=TE$+
STRING$(32-LEN(TE$),32)
```

The problem occurred because the program stores the book titles in left justified, blank filled, 80-character strings. Unless the search string is also blank filled, the program won't be able to locate any titles.

I love your magazine. It is the only one I buy for the CoCo, and it is more than enough. Keep up the good work!

Howard Hartsfie
Tallahassee, FL

Editor:

In the June 1983 issue of *Rainbow* on page 68 there is a small program called *Disk Display.* I found one very good use for it; I examined the sectors of Adventure games such as *Sands of Egypt* by Radio Shack, which as you know, does not contain the commands you need to use. By searching through the sectors, the commands will become present. This program gave me some help in finishing *Sands of Egypt.*

Mike Shor
Morganville, NJ

ADVENTURE WIZ

Editor:

I have answers to the following adventure games: *Calixto Island, Black Sanctum, Raakatu, Pyramid, Bedlam, Madness And The Minotaur.*

Anyone needing help can send me a S.A.S.E. with their questions and I'll send them the solution: Rt 2, Box 28 EF, 32536.

Joseph D. Fabian
Crestview, FL

HIDDEN HINTS

Editor:

Reading Tom Szlucha's article on the hidden commands of the MC-10 (September 1983), I thought it might be of interest to your readers to have the following information:

> USR vector: Store 16-bit address at $4216 (16918) Keyboard input character: use JSR $F865 (returns in ACCA) Output character to screen: Load ACCA with ASCII, use JSR $F9C6

I have not figured out the parameter passing for USR yet, but this works well to transfer control to an assembly language routine. There is a JMP instruction at $4215, followed by a 16-bit address that is the user vector. This points to an error message during system initialization. Incidentally, inspection of the ROM at address $E000 (57344) will reveal a list of 16-bit addresses, followed by the actual commands (ASCII), followed by more addresses. The USR function is the third entry in the vector table ($E006).

There is a jump table in RAM around $4215 that has the user adjustable vectors for all the interrupts in the system, including SWI. These can be very useful for figuring out how the MC-10 works.

A couple more key addresses: the MC6847 video controller is located at $A000 (40960). You can make all kinds of neat sounds by flipping bit 7 of $A000 up and down (this is "connected" to the speaker). Be careful not to change the other 7 bits of $A000 as these are the control bits for the 6847! Also, the "powered up" byte in the MC-10 is located at location $EA (234). On reset, the MC-10 looks in $EA to see if it contains a $55 (85). If not, it reloads all defaults and clears any program that might have been in the unit. If $55 is found there, it does a "warm" start and simply clears the screen and prints OK. I'm sure it also reinitializes the system I/O on either type of reset.

Hope this information is helpful.

Bruce R. Knox
President, Microdimensions
Willoughby, OH

Editor:

I have found out how to start on higher levels in *Donkey King.* To do this you must *POKE 12889,10* to start on the rivets level, *POKE 12889,2* to start on the elevator level, or *POKE 12889,18* to start on the conveyor belts. All of these are on a high skill level. If you would like to experiment with this you can also change 12889 to other numbers each of which will have some effect on the game. I would also like to say that I believe *the Rainbow* is the best magazine for the CoCo. Keep up the good work.

Robert Williams
Moore, OK

Editor:

In the August *Stained Glass* article, Wes Fauske asked why his "SG24" mode used only 3K instead of 6K of memory.

When he set up his graphics he got into the GIC mode instead of the SC24 mode. SC24 uses 6K and GIC uses 1K of memory.

By not setting the "Video Control Register," Wes did not get a true GIC but got caught between GIC and the "TEXT" mode.

Change the last number in line 25 of his program from 128 to anything between 65 and 90. The screen will be filled with 512 TEXT letters between A and Z. You may poke address 13019 with these values if you saved the program in machine code.

Michael B. Kromeke
Albuquerque, NM

WHY NOT SAVE/LOAD?

Editor:

I am writing about Dan Downard's *Rainbug Monitor*. I have a suggestion for him. I would like to SEC disk save/load machine language programs. There is disk memory examine/change, so why not save/load? How about a disassembler, too?

Andy Janjigian
Westwood, MA

Editor's Note:

Disk BASIC stores programs at a higher location in memory than Extended BASIC. Sometimes memory conflicts occur. A solution is to make the machine language program reside at the top of memory. Software author's for *Rainbow* are encouraged to write the ML portion of the program in PIC (position independent code) and also locate the program near the top of memory so that it can be protected by a *CLEAR* command.

MULTILINGUAL COCO?

Editor:

When I was reading the September 1983 *Rainbow* edition, I saw one article that really made me wonder. The article was about the new CoCo and its special OS-9.

I was wondering if I could get the OS-9 and the thing that lets me use different languages for my old CoCo. My computer system consists of a 64K CoCo, one DOS 1,0, one cassette, and an Okidata 82A printer.

Denny Reinhardt
Orofino, ID

Editor's Note: OS-9 will operate on any 64K CoCo and *it is* the system that allows higher level languages.

FELLOW NEWSLETTER

Editor:

As I read through your magazine, which I happen to enjoy very much, I often stumble onto the fact that *the Rainbow* magazine started as a two-paged photocopied newsletter. I am sure that a few years ago you never dreamt of a three-hundred page fully publicized major magazine. I commend and compliment your magazine but I also would like to ask for help. I am starting a newsletter and I have not had much luck in syndicating it. I feel it is a good little newsletter (only a few pages) and I think it could do well for other people such as column writers, unsung game programmers and columnists. I ask you to print this letter so people will send in their contributions of these items. If successful, I would like these people to write for my newsletter permanently and maybe someday it will turn into a real magazine. If you would just send a self-addressed envelope (to 8 Paprota Ct, Parlin, NJ 08859), you will receive our first newsletter with your articles and submissions. I love to write and I love my Color Computer and wish you people would send your submissions. The small businessman is welcomed and for you who would like to advertise and have a program to sell, please write me. I can't describe the willpower and earnest of our staff (only 12) but we have had no luck reaching people who would like to contribute.

Matthew Kurzawa
Parlin, NJ

KUDOS

Editor:

I just received my September issue of *Rainbow* in the mail. I'm just "itching" to read the darn thing; but, I had to write this Thank You note, first.

After my notification that I had not received the issue, you sent the issue priority mail to me! Fantastic!

This example of customer interest is exactly why *the Rainbow* is the number one CoCo magazine!

Ronald E. Beall
Ozona, TX

Editor:

I have really learned a lot through the pages of *the Rainbow*, especially the "BASIC Training" and "Bits and Bytes of BASIC" series, as well as typing in the programs. I am another subscriber who reads each issue from cover to cover.

Keep up the good work!

Lee Deuell
Shell Rock, IA

THE BEST DOWN UNDER

Editor:

I suppose there can never be such a thing as an overload of genuine praise. Therefore, I send my compliments to both you and *our* magazine. Today, out of curiosity, I purchased some other Color Computer magazines from a Sydney newsstand . . . there is no comparison with *the Rainbow!*

Rob Walls
Sydney, Australia

Editor:

Well, it has been two great years since I bought my CoCo, and with the great articles in your magazine I have improved my original 4K to 32K and have a new keyboard.

I think you have the best magazine in the world, without a doubt.

George Kaakee
Royal Oak, MI

Editor:

I would like to comment on your magazine. It is the best magazine for the CoCo around. No other magazine has the great graphic programs that you do. Since I've been buying your magazine, I feel the program called *Roach* is my favorite. Whenever people come over, they ask to play the game. Sometimes I wonder if they come over to see me, or to play the game. Keep up the fantastic job, and I'm looking forward to more great programs!

Dale Westmoreland
Lannon, Wisconsin

LEARNING BY R.O.T.

Editor:

Rainbow On Tape is a "good buy," but the teaching advantage is what I find to be the most advantageous. I read almost all the articles in all of the computer magazines. Or that is, the CoCo computer magazines. In doing this, I have not had a lot of time left over to type in printed programs — especially the ones that are more in the "learning" category. However, as I have *all* of *the Rainbow* programs on tape, I can *RUN* every program. That way I can learn the lesson that is presented and also see the program in operation.

Please do make every effort that you can to get the most — no all — of your readers subscribing to *Rainbow On Tape*.

Russel M. Hokanson

PEN PAL SERVICES

Editor:
I am starting a service for CoCo owners who want a pen pal. You can send some information about yourself and what your uses are for your computer. Please include a 20 cent stamp and 20 cents to cover costs. We will send you the address of someone compatible with yourself. Mail it to Lee Ave., 32312.

Scott Santarone
Tallahassee, FL

A COCO MATE

Editor:
In response to William Burns' query in the September *Rainbow*, the Mannesmann Tally MT160 printer is an excellent mate for the CoCo. It interfaces directly (I run at 2400 Baud). With its multitude of features, including graphics, EPSON and Daisy Wheel code support, large buffer, proportional spacing, etc., it's a dream. Any CoCo owner needing cable fabrication or setup information can send a S.A.S.E. to: Hollow Tree Soft., Rt. 10 Box 388, 29640.

Kevin Davidson
Easley, SC

MODEM TALK

Editor:
I have just purchased a Modem I from Radio Shack.

Now I am interested in learning from others about the modem and sharing information and conversation with other modem owners. I can be reached at (512) 442-6317.

David Karam
Austin, TX

INFORMATION PLEASE

Editor:
My husband and I think your magazine is the greatest! There are just so many helpful articles in it.

We have a Radio Shack TRS-80 CoCo Extended BASIC and have been having trouble with the *Raaka-Tu* game. We can get all of the way through up until we get to the rectangular room with the carpet over the pit. You cannot jump over it and there is no room to go around but you need to get to the door on the other side. We would appreciate it very much if any of your readers have a solution. Please send answers to: 4141 Hamilton-Eaton Rd. #57, 45011

Mrs. J. L. Whitaker
Hamilton, OH

Editor:
I need help solving *Sands of Egypt* and *Pyramid*. I can't get past the great hall in *Pyramid*. Also, I can't find the pool, camel, and pyramid scene in *Sands of Egypt*. If anyone can help me, please contact me at 2225 Hanley, Odessa, TX 79762.

Paul Pettit
Odessa, TX

Editor:
I purchased the C. Itok Prowriter 8510 printer a couple of months ago. It's a great printer when interfaced to the Color Computer with the BOTEK Parallel Printer Interface. I just POKE 150,1 to set the computer Baud rate to 9600 and set t
switch to the same setting. This combination works like a charm. This great printer should be looked at by anyone in the market for a new printer. I highly recommend it.

I would like to know if anyone knows of a screen dump program that works with this printer the 8510 uses different control codes than the Radio Shack printers, so the screen dump that they use doesn't work at all.

I've been with you since Feb. '82, and have watched you grow from 31 pages to over 300. I look forward to each exciting issue in my mailbox each month. Keep up the good work.

Charles M. Thonen
Ft. Greely, Alaska

Editor:
I have a TRS-80 Color Computer, 32K, which I use a tape recorder with. I just purchased a DMP-120 printer from Radio Shack. I would like to know how I can print a certain number of lines so when I use fan fold paper and come to the end of the paper, I can't have half a line on one sheet and the other half on the next sheet? Is there a statement I can type in that skips so many lines and types so many and so forth? Thank you.

Sam Gurrera
Parma, Ohio

WHICH WAY TO BASIC?

Editor:
I have been using my *EDTASM+* cartridge to disassemble the *CLOAD* command. I have found everything I need for my program but the simplest thing . . . where it returns to BASI . Will someone please help!? My phone number, for those of you who live in Austin, is 442-6317. My address is 1809 Dexter, 78701.

Thank you.

David Karam
Austin, TX

LET'S BOGEY

Editor:
I have only been a subscriber for about 6 months, but really enjoy your magazine.

Since golfing season started I've been looking for a program to handle our clubs' golf handicapping on my CoCo.

Any help in finding this program would be a big help for next season. Thank you.

F. Warren Crowley
Whiting, NJ

Editor:
I have *Telewriter-64* and get tired of entering the same formatting information each time I want to print something. I have added the needed *POKE 150,1* to line 0 and the appropriate *POKE* from page 33 of the *Telewriter-64* Reference Manual to line 210 of "U" to eliminate the sparkle but this is as far as I have been able to get.

A letter to Cognitec was of no help. Is there anyone out there who can figure out how to change the other default format numbers? As I told Cognitec, "I don't wish to meddle with a well-programmed piece of software but not having to do an operation (the same way) every time I wish to print something seems a legitimate reason to change."

"Spike" Worden
Memphis, TN

MINOTAUR MADNESS

Editor:
I have the game *Madness and the Minotaur* and need lots of help. If anybody can give some tips or the solution, send it to: 11 Page Drive, 71203.

Also, you have a great magazine. Keep up the good work!

Edwin Bruce
Monroe, LA

Editor:
I need some help solving *Madness and the Minotaur*. My problems are getting out, obtaining spells and killing monsters. If you have any information on my problems, send it to me at 139 Park Street.

Tom Spinoro
Savanna, GA

Editor:
I am interested in obtaining new languages for my 64K CoCo with disk drive. I would like to get Pascal, Cobol, and Fortran. What would my best bet be? Radio Shack's new OS-9 or maybe a FLEX? Also, where could I get quality software for it?

If anyone has any suggestions, please contact me. My address is: 19 Magnolia Ave, 07834.

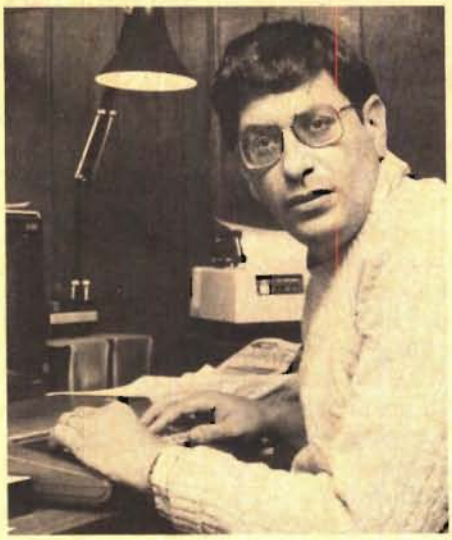
Mark Charney
Denville, NJ

POSTAL POKE

Editor:
We have recently learned that several people who ordered SpectroSystems' *Fastape* program for high-speed cassette operations following the review in your July issue had their orders erroneously returned by the post office. The problem with the postal service has, we hope, been straightened out, and we'd like to reassure *Rainbow* readers that we are indeed at the address that was given: 11111 N. Kendall Drive, Suite A108, 33176.

Arthur J. Flexser
SpectroSystems
Miami, FL

PRINT #-2,



I admit I realize that some of you will be reading this somewhat late to do much about it, but just in case you are not, there is still time to make RAINBOWfest in Fort Worth Oct. 14-16.

I think that this is going to be one truly great show! And, perhaps more important, we *have* gotten the word that there will be tours of the Tandy CoCo manufacturing plant during the RAINBOWfest. If you want to see how, and where, CoCo is "born," so to speak, RAINBOWfest in Fort Worth is the place to be!

The tour will be at "cost" — just what it will cost to rent the buses to take us from the Hyatt-Regency Fort Worth to the plant, a total of $3 per person. But the space is limited. So, if you wanted to make it to RAINBOWfest but needed a reason until now — this is a great one.

I'm pleased, too, that we have what I consider to be an outstanding lineup of people for seminars at RAINBOWfest. Tops on the list is Mark Yamagata, the new Director for Computer Merchandising —Personal Computer Products at Radio Shack. Mark has just been appointed to his new post and this will be one of the first public appearances he will be making. We feel especially honored that he is choosing to keynote our "CoCo Community" Breakfast. Be sure to put the breakfast Saturday morning on your RAINBOWfest agenda.

Mark is not the only "name" we will be having. Some of the CoCo world's most popular programmers will be on hand; as will a number of our *Rainbow* columnists. We think that RAINBOWfest will be not only an excellent exposition, but an outstanding opportunity to learn, too. With people like Don Inman, John Gabbard and Steve Blyn there, how can you miss out?

And, in case you missed the notice in the advertisement, we've been fortunate enough to set our keynote speaker for the second RAINBOWfest of the season, at Long Beach, Calif., in February. Bob Albrecht, the reknowned columnist and author, will keynote *that* "CoCo Community" Breakfast.

By the way, someone asked me the other day why we're having four RAINBOWfests this year. The answer is a simple one: We'd like to have the shows in locations that will be relatively easy for you to travel to. One of the things which impressed me about the first show last year in Chicago was the long distances some of the RAINBOWfest-goers came to attend. Thus, I suppose, was born the concept of "regional shows." Make no mistake about it, having four RAINBOWfests is far more costly, in terms of both time and money, than just one or two. But our aim, overall, is to make it as easy as possible for the most number of people to participate as possible.

No, I'm not saying we're doing this just out of the goodness of our hearts. But I *am* saying that, frankly, we will make less money by doing four shows than by doing just one or two. Still, we should do better than break even, and we do want to "spread the word" about CoCo as widely as we can; making it possible for as many to attend as inexpensively as they can. Thus, regional RAINBOWfests.

I'm especially pleased with some of the longer programs in this month's issue and, although I usually let Managing Editor Jim Reed write about the contents of a particular issue, I'd like to "brag" on these special ones.

As most followers of *the Rainbow* probably know by now, one of the abiding interests we have had has been in da communications. And so, this is our Data Communications Issue. That, in and of itself, is something important, but what's more important is the fact that we wanted to do something special for this special subject.

Rainboard was the answer to this special need. It comes from the talented Dr. Lane Lester who has, over the past couple of years, made a number of significant contributions to *the Rainbow*. None of them is more significant than is this program, which, I feel sure, will open up a whole range of communications possibilities to thousands of you.

While expensive from the standpoint of telephone calls over long distances, communicating with local bulletin boards is a way for CoCoists to easily ban together. Imagine! Now, in any local telephone calling area where there is more one Color Computer, a person has the means to communicate without missing messages and through the case that a computer can bring. Maybe, just maybe, we should have saved Lane's program until the January issue, called that one the "George Orwell Issue" and presented *Rainboard* — something that is truly 1984-ish.

If data communications bore you, though, just consider this one: You can easily run *Rainboard* in your own home and use it as a means to leave messages and the like with other members of your family. You don't even need a modem to do that. Lane, thanks for a fine program that I am sure will have many singing your praises.

The other two special programs are our two Simulation Contest winners. What could be more appropriate heading into a presidential election year than *Election '84*? And, although the judging was completed before Korean Air Lines Flight 007 was shot down by the Russians, what could be more timely than *WarGame*? It is,

indeed, sad that we must live in a time when a nation can so calmly slaughter innocent men, women and children as the Russians did last month.

Special best wishes to our two winners. One will receive an Epson FX-80 printer and 4K buffer from JARB Software and the other, a Disk Drive 0 and controller from us. Other winners get prizes from Spectrum Projects, Custom Software Engineering, Computer Island, Tom Mix Software and Classical Computing.

And, our thanks to the many who contributed to our Simulation Contest. There will be another, after the second running of our Adventure Contest, which begins with this issue.

And, on to the rules of the Adventure Contest itself. You must submit an original Adventure program and certify to us that it is your own, original work. The program should be on tape or disk and can be either a graphics or a non-graphics game.

All entries become the property of Falsoft, Inc., and none can be returned. Deadline for submissions to the contest is March 1, 1984. Winners will be announced in our special Adventure issue. As usual, there will be a list of top prizes. Duplicate prizes to be awarded in case of ties.

Please include full documentation, along with the *solution* to the Adventure. Entries should be addressed to *Adventure Contest Editor, Rainbow Magazine, P.O. Box 209, Prospect, KY 40059*. Last year's contest was just super and we expect this year's to be even bigger and better. Watch future issues of *the Rainbow* for a full list of prizes.

I received a letter from one of our readers the other day, asking whether we could show photographs of our office. The writer expressed interest in seeing what our "office" looked like.

I'd like to know what it looks like, too. As many of you are probably aware, we have been expanding rapidly for some time. So, the reason *I* wonder what things look like is because between the people, furniture and computers, it is almost impossible to see the "office" itself. We began on our kitchen table, moved to 1200 square feet of renovated basement and then to 2000 square feet of office space in the Prospect Point Shopping Center nine months ago. We choose the particular location that we are now in because it was right next door to the Post Office and because of an interest in helping us on the part of both the real estate manager for the complex and the people who own it. When things began to get crowded (about the day after we moved in, it seemed), we wondered where we might grow into next.

Happily, as you read this, the problem has been solved. We now occupy *both* sides of the Post Office and have an additional 2850 square feet from which to continue to bring you *the Rainbow* (and *PCM — The Portable Computing Magazine*) each month.

Splitting up means we had to split something up, of course. So, we "left" our editorial and design staffs in the old space and moved Customer Service, Accounting, *Rainbow On Tape*, Advertising, Shipping and our other "business" departments to the new location. For the first time, editorial and design finally have room to do all the things they need to, and the business side has enough space to really operate and file things.

One of the things which has made this all possible is, as usual, Radio Shack. Basing things on the excellent success we've had with their computers (the "business" part of the business is all on Radio Shack computers, too, and yes, we do all the subscription servicing and so on ourselves), we opted for a Radio Shack telephone system. Thanks to the expertise of a great many people, we may be in two separate offices, but we're tied together by one phone system where — if necessary — everyone can talk to everyone else. We can conference calls, use speed dialing and so many other things this column might begin to sound like an ad written in Tandy Center if we kept on. We won't. But, *if* you do have occasion to call us, you'll be in good hands!

No, our telephone number will not change, nor, for all practical purposes will our address. An added convenience for callers is that we have almost doubled the number of telephone lines we had previously, so there won't be as many busy signals.

If you get the idea we're proud of our new offices, you're right. And, by the way, we welcome visitors. So, if you are ever in the Louisville area, please stop by. We'd like to see you in person.

As you know, we're about a month ahead on these columns and each issue of *the Rainbow*. So, we have not put it all together yet, but please be aware that, as I mentioned last month, we will have a subscription price increase. It will most likely be effective January 1, 1984. As we did so a year ago, we offer all present subscribers an opportunity to extend their subscriptions at the present rate ($22 in the United States, U.S. $29 in Canada and Mexico, and U.S. $57 surface to other countries, U.S. $85 air mail to other countries). Right now, we're working with our printer to see what it will cost to publish next year and expect to formally announce our rate increase next month. Meantime, if you would like to extend your subscription by one year at the old rate, now is the time to do so.

While on the subject of subscriptions, let me point out that we do have some very nice gift certificates available for holiday giving. Just ask that a gift subscription be sent when you write.

And, this was mentioned to me some time ago as a suggestion: A really nice gift you can give for all of the next year is a subscription to *the Rainbow* to your local school or library. Not only is it a way to help your school or library, but the subscription cost is tax deductable as a charitable contribution. It would be a nice present, and may help someone get started in computers the right way — with a CoCo!

—Lonnie Falk

BUILDING NOVEMBER'S RAINBOW

Our Data Communications issue . . . Introducing the Rainboard BBS . . . And, our Simulation Contest Winners . . .

We get far more mail here at *the Rainbow* than at any other publication I've been associated with. And, while only a small portion makes it into the *Letters to the Editor,* or the new letters sections in both the CoCo Clubs section and the new Reviews section, we do take our readers' criticisms, suggestions and observations to heart. One reader from North Carolina recently sent in his personal list of concerns, closing with: "Keep On Keepin' On, But Do It Better! And Hurry The Process!!"

Well, W.J.F., we're trying to do just that and, yes, we have modified *Rainbow On Tape* so that you no longer have to turn off the machine after running the menu. That was an oversight on our part and we appreciate your calling it to our attention. Similarly, your other concerns and those expressed by other readers are being attended to as quickly as we can do so, but do have patience because we have a long lead time and thoughts expressed today may require months to implement.

This month we focus on data communication, computer-to-computer communication, a concept that sounds speedy, but can be painfully slow at 300 Baud. Still, the mostly untapped potential of data communication is fascinating. For instance, with *Rainbow* Technical Editor Dan Downard's *Remote,* a machine language driver program, you'll be able to run your CoCo by remote control. With our list of computer Bulletin Board Systems, you'll be able to "talk" to other computer users anytime night or day. With Alan Davenport's *Master Messenger,* you'll be able to format "E-Mail" before you even log on services such as The Source and CompuServe. And, as our special feature this month, we have *Rainboard!*

Rainboard is a complete BBS, or Bulletin Board System, that our favorite biologist, Dr. Lane Lester, has designed to work on a single disk drive with an "intelligent" modem. And it's yours to have and to use for the price of this issue. Now, the chance to be a "SYSOP" is within range of thousands of us. Just boot it up and wait for the calls to come in. Ma Bell should pay us a royalty for the increased business we predict this will generate. *Rainboard* isn't The Source, but it *is* user modifiable and can be your source for lots of fun and new-found friends. Enjoy!

As always, we have a diverse mix of articles and programs in addition to our data communications theme. These range from a lesson in "disk washing" — don't take that too literally — and disk calibration to the "glowing" love story of "The Laserworm and The Firefly." Paul Hoffman has another X-Pad feature and Tommy and Gail Pollock offer a musical "Bach to BASIC."

Finally, of the several prize-winning entries in our Simulation Contest, our two top awards go to Bob Tyson for *Election '84* and to Tom Weber for *WarGame.* Both Simulations are exceptional, in length and in reward, as well as in timeliness and playability, so don't miss them. No, you don't have to run for President before playing *WarGame.* Simulation Contest Chief Judge Charles Springer offers an overview of the contest and reveals all the other winners, too, in his judges' report.

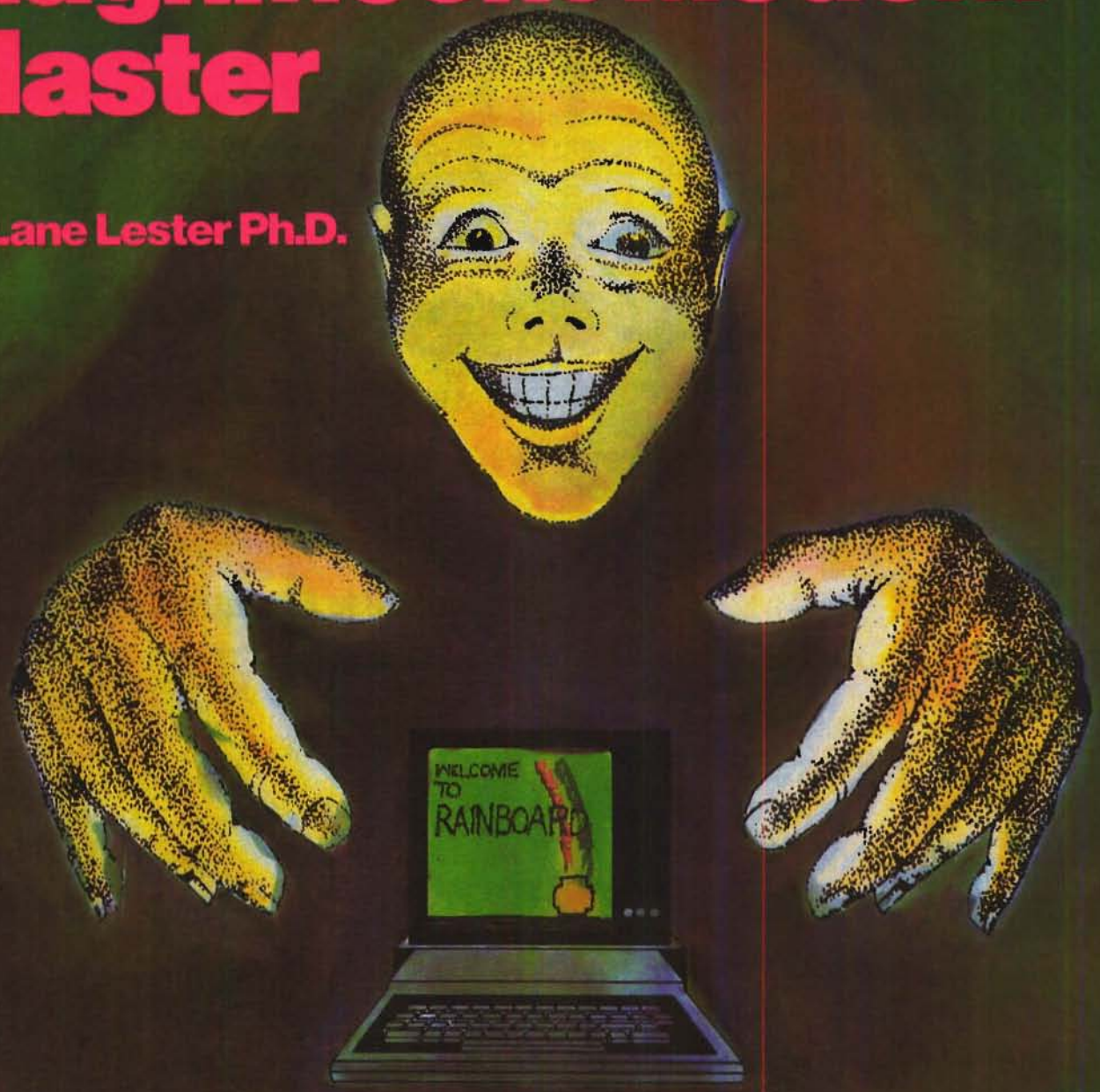
So, we keep on keeping on, with an eye to doing it better and a pledge to hurry the process. You keep the letters coming and we'll respond monthly with a *Rainbow* that weighs well over a pound. A full year's subscription is still just $22, less than half the cover price. We do get a lot of mail, but we want even more. That's why we, having just expanded again, now occupy office space on both sides of the local post office.

—Jim Reed

Now! After a lifetime of lusting for power, prominence, charisma, clout and godlike celebrity, you, too, can become a . . .

Supreme Sysop and Magnificent Modem Master

by Lane Lester Ph.D.



Have you bought your CoCo a modem yet? Have you sampled the joys (and the telephone bills) of roaming through the Network Nation? All over the country there are BBS's (bulletin board systems), computers waiting to take your call, systems dedicated to the interests of Color Computer owners. Here in the small town of Lynchburg, Va., there are very few Color users as dedicated (euphemism for "freaked out") as I, and communications capability allows me to reach out with telephone tentacles to like-minded individuals around the country. Some of my most valuable exchanges of information have come from Kent Meyers in LeRoy, Minn., by way of the Color Connection in Elgin, Texas. (By the way, Kent has supplied much of the information appearing in *the Rainbow's* listing of BBS's elsewhere in this issue.) In addition to your modem, you'll also need a terminal program, several of which have been reviewed in *the Rainbow*.

But this article is really aimed at those who have already come to appreciate both the enjoyment and edification that comes from communicating with other Color users via the bulletin board systems supporting our machine. To many of us, the SYSOP (System Operator, manager of the bulletin board) is a godlike creature, sitting like a spider in the middle of its web. The mighty SYSOP receives our transmissions from everywhere, answering our questions, and controlling what messages, files, and programs are available to us lower sorts. The aspiration to be like God led to Satan's downfall and many of us have lusted in our hearts to be a SYSOP, with all the adoration and power that accompanies that position.

I must confess to you that I have long had a secret desire to operate my own BBS. Of course, one very serious hindrance to such a project is that while the computer is running the board it can't do anything else. For the most part, only those who have an extra computer to spare can be a SYSOP. One day it occurred to me that since the CoCo in my office at school is alone all night long, it could run a BBS at night! Then I looked at the prices of bulletin board software, almost all in the neighborhood of $100 an up. Another blow to my ambition. One day I purchased for $19.95 a very interesting machine language program, *RemoTerm* by Star-Kits. I could load and execute this program in my office CoCo, go home and use my home CoCo by telephone to load and run programs in my office. It occurred to me that I could use *RemoTerm* to run a BBS; all I had to do was write a BASIC program to manage the system.

RainBoard, "With a Rainbow of Color and a Pot of Gold in Good Times," is now a reality, and I'm having a great time. Perhaps you would enjoy operating your own board; the following programs will allow you to find out without a great expenditure in cash, only in time. As far as hardware is concerned you'll need a 32K CoCo with Disk BASIC, one disk drive, and an auto-answer modem. A price breakthrough for this last item came with the development of the *Novation J-CAT,* which is discounted for a little over $100. But as important as is the right hardware, it's the software that makes a BBS the special thing it is, so the rest of this article will be devoted to describing the use of the programs that follow.

(Lane Lester is professor of biology at Liberty Baptist College and holds the M.S. in ecology and the Ph.D. in genetics.)

There are four programs accompanying this article which are included on *Rainbow on Tape*. *Remote* is the BASIC program that I run just before I go home each evening. It first loads *RemoTerm*, and then changes it to disable the callers' [BREAK] key. The modified *RemoTerm* is executed, and finally *RAINBORD* the BBS program, is run. This version of *REMOTE* also sets some parameters on my *Hayes Smartmodem,* and would have to be changed if you have a different unit. A timer turns off the system at midnight and then turns it on at 7:30 a.m., so that it's warm and waiting when I get to the office. *SYSOP* is the first program I run, displaying the activity on the *RainBoard* the previous night. It has some other functions that are self-explanatory and aid in the maintenance of the board. *GRAFMESS,* which I downloaded from the Color Connection and modified considerably, allows me to create the color graphics screens that I believe add a nice touch to the *Rainboard*. *RAINBORD* supports the following BBS functions:

Help with BBS functions,
Bulletins of interest to callers,
Scan message subjects for possible retrieval,
Read messages,
Leave messages,
Programs available for downloading,
Text files on various subjects,
Art gallery of graphics
Member list
Goodbye — leaves the *RainBoard* ready for the next caller.

Perhaps the most logical way to describe the use of *RAINBORD* is to relate my comments to particular sections of the program. Line numbers will be shown in parentheses. This should both help you see what's going on and aid you in customizing it for your own use. An immediate jump (20) to the bottom of the program avoids the notorious *PCLEAR* bug. The Lower/Upper Case Input Routine (3) is called at various times in the program and allows the caller to use either lower or upper case responses to prompts. Single letter responses produce a colored stripe across the screen which provides part of the *RainBoard's* promise of "a rainbow of color." Three arrays are dimensioned (70), DISPLAY$ (for graphics, 15/picture), NAME$ (BBS members), and TEXT$ (for message entry). Next (80) I load the three graphic screens that I'm currently using which were created with the graphics message program *GRAFMESS*. The first two are displayed in lines 150 and 390, respectively, and the third is in the Art Gallery (1220). If you add pictures to the Art Gallery, you'll need to change in line 1240 the number 45 to 15 more for each picture. The list of current members of the BBS is then loaded (100). Because the graphics and member list are displayed often, they are kept in RAM to minimize disk use. The disk file of members is, however, updated whenever a new member joins the board.

RAINBORD refers to disk drives either by default or as D1. By making all of them 0 (110) only one drive is required, and the only limitation is on the number of text, program, and message files that can be online. Once way to maximize

space is to keep the four BBS programs *RAINBORD, REMOTE, SYSOP,* and *GRAFMESS* plus the graphics files WELCOME, HELP, and those in the Art Gallery on one disk which is only used at the time the *RainBoard* is started. All other files can be on a second disk which stays in the drive as long as the BBS is online. Text files always go on the default drive (0), but because I am blessed with a Radio Shack drive 0 and a double-sided MPI drive wired as drives 1 and 2, my own version of line 110 reads: D1=1: 1$="1":D2=2:D2$="2."

When someone calls the *RainBoard,* program line 130 is waiting for a carriage return [ENTER]. The caller is given a chance (140) to request 7 or 8 bits , which will determine at various places in the program whether text or graphics will be displayed. For callers who are using the smart terminal program *ColorCom/E* the CHR$(27)"G4" sets the program to 8 bits. To speed logging on, the caller is asked for initials only (170) which are checked against the member list (190) and if a match is found, the caller is asked for verification (200). November callers go through the "New Member Signup" section (220-340), while for members the message files are checked and any new messages addressed to the caller displayed.

You'll notice throughout the program that a string, ACT$, is constantly growing. This is the record of the caller's ACTivity which is recorded when the caller logs off. FL is another variable whose purpose may not be immediately obvious. It serves as a FLag to allow the program to hang up on at least some mischief-makers. CHR$(13) is used where commas would ordinarily do so that callers with other than 32-character displays will see a decent screen. The last step in new member signup is the display of the text file "NEWMEMBER" (340) which provides information for beginners.

The Main Menu (350) consists of only a single line to speed communication. If a person responds with an invalid character, an "H," or simply a carriage return [ENTER], an expanded menu is displayed (380-440). After they are selected, most of the individual functions have available complete help files on the disk. As each function is selected, the *GOSUB1300* keeps track of both the type and amount of the caller's activity.

The "Scan Messages" function (480-560) next needs comment. Each message occupies a separate granule on the disk, and the scan function displays both the filename and extension. Filenames indicate the subject of the message and extensions the initials of the intended recepient. Messages to "All" have extensions such as "A09" to indicate the month entered. The caller can request Help, a Complete scan, Initials to scan, or Return to Main Menu.

The "Read Messages" function (570-720) offers similar options with the ability to retrieve selected Subjects or Initials. A directory search subroutine (640-710) serves all options as well as the initial message search when a member first calls the *Rainboard.* The ability to create private messages is provided when the subject specified is "PERSONAL" so that only the intended recipient can retrieve them (670). When a caller reads one of his or her messages, an asterisk replaces the middle initial in the file extension (690), making possible the display of only new message while leaving old ones for others to read. An opportunity to reply is immediately provided the recipient (700).

The caller's options in the "Leave Messages" function are different in that three letters are expected (740). A lengthy series of checks and prompts insure that only valid disk file specifications are given by the caller, and that duplicate filenames do not erase other messages. Because most CoCo terminal programs have word-wrap prevention, a 63-character line length for those who print their messages after going offline. A "." printed above each line helps the caller keep track of the line length, but if he or she exceeds 63 characters an error message is displayed and the line truncated to the proper length (900-910).

The remaining functions do not require much comment. For both BASIC programs to download and text files to read, the caller can get a List or Descriptions (950 and 1090). The Art Gallery (1220) is a nice idea that I lack the artistic talent to fully exploit. The Goodbye function writes the caller's activity to a file (1280), says a personal goodbye (1290), and hangs up the *SmartModem* (1300-1320) before jumping back to the beginning for the next caller. If you use the *Novation J—CAT*, you can delete lines 1300-1301; it will hang up after the caller.

The Display File routine (1330) is called whenever an ASCII file of any type is to be transmitted to the caller. This includes messages, text files, and programs. The Activity Length Test (1360) keeps track of each caller's use of the *Rainboard* and helps to prevent someone's monopolizing the system. If a caller is having trouble using the BBS, you can usually see what's wrong when you display the activity file using the program *SYSOP.* Several places in *RAINBORD* the membership list is checked (1400), and each graphic screen created by *GRAFMESS* is loaded by the last routine (1420). In line 80, the values of Y and Z determine the location of each graphic in the single array.

Rainboard Procedures

The following is a step-by-step list of things to do to get the *Rainboard* up and running:

1. Use *DSKINI0* to erase and format two disks. We'll call them Disk 1 and Disk 2 in the following steps.

2. Put on Disk 1 the four BASIC programs *REMOTE, RAINBORD, SYSOP,* and *GRAFMESS*. You can either type them in from the magazine, get them on *Rainbow on Tape*, or best of all, send me $20 for a disk also containing supporting text and graphics files. Also *SAVEM* to this disk either *RemoTerm* or the machine language program written by *Rainbow* Technical Editor Dan Downard and appearing elsewhere in this issue. Edit *REMOTE* to contain the name of the program you decide to use.

3. Run *GRAFMESS* (described below) and create the graphic files WELCOME, HELP, and any that you would like to display in the Art Gallery. Save these to Disk 1.

4. Run *SYSOP* (described below) and select menu option 6 to create the MEMBERS file. Save this to Disk 2.

5. On Disk 2 place the text files (described below) TEXTDESC, PROGDESC, BULLETIN, NEWMEMBR, SCAN, READ, LEAVE, PROGRAMS, and TEXT. Also add any text files that you want callers to be able to read with the Text option of *RAINBORD*.

6. Save on Disk 2 the BASIC programs you want callers to be able to download to their computers. These must be in ASCII format and should be given extensions as follows for lines 980-990 of *RAINBORD*

POU-Utility
POA-Application
POG-Game
P0O-Other types of programs.
Example command: SAVE"MILE MON/POA",A

7. Now you're ready to go! Turn on the modem, and be sure that it, rather than your printer, is hooked to the RS-232 outlet. Place Disk 1 in your drive and *RUN*"REMOTE." After the files on Disk 1 are loaded, you will be prompted to remove it and insert Disk 2. Pressing [ENTER] will load the MEMBERS file, and you will receive the message, "RAINBOARD IS READY TO RECEIVE!"

8. The *Rainboard* is now waiting for callers, and requires no further attention from you.

9. Each day at a time convenient to you, you should first "call" the *Rainboard* yourself, either from another computer or directly by pressing [ENTER] on the *Rainboard* keyboard. This will allow you to read messages addressed to you and to reply to them. Then, turn the computer off and back on, insert Disk 1, and run *SYSOP*. Use either menu options 1 or 2 to view the activity file. If a caller seems to have had trouble, you may want to leave him or her a message about it.

10. Use *SYSOP* MENU option 3 to kill the old activity file. Periodically, you will also want to use option 8 to remove messages that have been read by their recipients.

11. Return to step 7.

Text Files

The subject of text files deserves further comment. There are two types of text files used by the *Rainboard,* one of which provides information to the caller on the use of the BBS and the other type on various topics of interest which can be read through the Text function of *RAINBORD* In this latter category, I currently have the following: DSKBASIC/073, a memory map of the new 1.1 Disk BASIC ROM; COCO BBS/083, a list of CoCo boards; PROGTIPS/053, programming tips; and FOR SALE/073, sources of good buys. The extension gives the date of entry, e.g. 093 =September, 1983. When a caller requests a list of text files, *RAINBORD* (1080) displays only the filenames that have numerical extensions. The following text tiles, all of which have the extension "DAT," are displayed at different times in the use of the BBS:

TEXTDESC — descriptions of the text files that can be read by the caller;

PROGDESC — descriptions of the programs available to callers;

BULLETIN — news of meetings, etc. Note that line 460 of *RAINBORD* has to be edited whenever you update this file.

NEWMEMBR — explains use of the bulletin board; automatically displayed when a caller first joins the *RainBoard*;

SCAN, READ, LEAVE, PROGRAMS, TEXT — help files for each function of the BBS. Just as the Help function explains each of the Main Menu choices, these files contain descriptions of submenus and other aid in the use of the particular function. As an example, the SCAN file is listed below.

Text files can be created most conveniently with any word processor program that is capable of producing ASCII disk files. Several of these are advertised in *the Rainbow*. If you don't already have one, the cheapest way to produce your text files is to *RUN*"RAINBORD" and use the Leave message function to create the files as messages. You'll have to enter your own initials as the recipient, but they can be *RENAMEDed* later with the date extension needed for *RAINBORD's* text file function. The example file SCAN will indicate the necessary characteristics of *RainBoard* text files. The line length should be set to 31 characters/line, and only capital letters whould be used. It is best not to use hyphens to divide words. Some callers will have other computers and will want to later print what they receive with a longer line length.

Disk File SCAN

```
THIS FUNCTION LISTS THE
SUBJECTS AND RECIPIENTS OF
MESSAGES ON FILE. YOU CAN THEN
READ ANY THAT INTEREST YOU. THE
MENU OPTIONS ALLOW YOU TO SCAN
ALL OF THE MESSAGE HEADINGS OR
THOSE ADDRESSED TO SELECTED
INITIALS.
MESSAGES DIRECTED TO 'ALL' MUST
BE SPECIFIED AS 'A' PLUS TWO
DIGITS FOR THE MONTH DESIRED,
E.G, 'AO8'.
OPTIONS AT THE PROMPT ARE:
H (THIS SCREEN
C (COMPLETE SCAN)
I (SCAN FOR INITIALS)
R (RETURN TO MAIN MENU)
```

GRAFMESS

The *GRAF*ic *MESS*age generator program is used for the creation of the graphic screens displayed by the *Rain Board*. Instructions for the use of *GRAFMESS* are displayed when the program is run. The user is prompted to select the creation of a new graphic or the editing of one already in a file. After indicating either disk or tape storage, a filename is requested. If one is editing an existing graphic, it is then loaded from the disk or tape. There is a delay while the graphic array is processed for editing, and then a blinking cursor indicates that the program is ready for keyboard input. If a new graphic is being created, the user is asked for a *CLS* value which clears the screen to that color.

The arrow keys are used to move the cursor to desired locations. To select a particular graphic character, the [CLEAR] key is pressed. Then two key presses are used to enter the hexadecimal value of the desired character. These values are listed on page 14 of the Nanos reference card. Each time the "@" key is presed, the selected character will be displayed. Pressing the back arrow also displays the character, and this allows rapid erasing of areas. Text characters are simply typed as usual. When the graphic is complete, pressing [SHIFT][CLEAR] will store both the ASCII file for *RAINBORD* and the binary file for later editing.

NOTE

The four programs, *RAINBORD, REMOTE, SYSTO-P,* and *GRAFMESS,* and the supporting text and graphics files are available on disk for $20 from the author, 413 Woodland Circle, Lynchburg, Va. 24502.

Listing 1

```
10 'REMOTE
20 CLEAR800,&H7F00:LOADM"REMOTER
M"
30 'Disable Break (Control-C)
40 POKE&H7F50,&H86:POKE&H7F51,&H
2A:POKE&H7F52,&H12:EXEC
50 'Enable SmartModem
60 PRINT"AT E0 M0 Q1 S0=1 S5=130
 S10=20"
70 RUN"RAINBORD
```

```
100.....0205
280.....03B6
430.....0739
END.....0921
```

Listing 2

```
10 'GRAFMESS: Graphic Message Ed
itor
20 'Original version from The Co
lor Connection (512)285-5028, ed
ited by Lane Lester
30 CLEAR5000:DIMLL$(15,32):GOSUB
370:AD=&H400:KEY$=CHR$(8)+CHR$(9
)+CHR$(10)+CHR$(12)+CHR$(64)+CHR
$(92)+CHR$(94):LS=&H80
40 ST=PEEK(AD):POKEAD,&H2E:J=1
50 K$=INKEY$:IFK$<>""THENPOKEAD,
ST:S=ST:GOSUB520:GOTO90ELSEJ=J+1
:IFJ<5THEN50
60 POKEAD,ST:S=ST:GOSUB520:J=1
70 K$=INKEY$:IFK$=""THENJ=J+1:IF
J<25THEN70ELSEGOTO40
80 ' Decipher Keys
90 K=INSTR(KEY$,K$):ONK GOTO170,
140,230,260,300,320,200
100 A=ASC(K$):IFA<64ANDPEEK(&H11
A)THENA=A+64ELSEIFA<64ANDNOTPEEK
(&H11A)THENA=LS ELSEIFA>96THENA=
A-96
110 POKEAD,A:S=A:GOSUB520:IFAD<&
H5FF THENAD=AD+1
120 GOTO40
130 ' Right Arrow
140 IFAD<&H5FF THENAD=AD+1ELSEAD
=&H400
150 GOTO40
160 ' Left Arrow
170 IFAD>&H400 THENAD=AD-1ELSEAD
=&H5FF
180 ST=LS:GOTO60
190 ' Up Arrow
200 IFAD>&H41F THENAD=AD-&H20
210 GOTO40
220 ' Down Arrow
230 IFAD<&H5E0 THENAD=AD+&H20
240 GOTO40
250 ' Enter New Shape Value
260 L$=INKEY$:IFL$=""THEN260ELSE
L=ASC(L$):IFL>96THENL$=CHR$(L-32
)
```

```
270 S$=INKEY$:IFS$=""THEN270ELSE
S=ASC(S$):IFS>96THENS$=CHR$(S-32
)
280 LS=VAL("&H"+L$+S$):GOTO40
290 ' Display Shape in Memory
300 ST=LS:GOTO60
310 'Write to Transmit File
320 OPEN"O",DV,F$:LL$="":LL=VARP
TR(LL$):POKELL,32:FORI=0TO15:LO=
&HOE00+I*32:MSB=INT(LO/256):POKE
LL+2,MSB:POKELL+3,LO-MSB*256
330 WRITE#DV,LL$:NEXT:CLOSE
340 'Write to Edit File
350 IFDV=1THENSAVEMF$,&H400,&H60
0,&H400:SOUND200,2:GOTO40ELSECSA
VEMF$,&H400,&H600,&H400:SOUND200
,2:GOTO40
370 'Instructions
380 CLS:PRINT"***GRAPHIC MESSAGE
 GENERATOR*** @ KEY DISPLAYS GRA
PHIC SHAPE.   CLEAR KEY TRIGGERS
 NEW SHAPE:"
390 PRINT"   ENTER HEX VALUE FRO
M NANOS      CARD WITH TWO KEY P
RESSES.","ARROW KEYS MOVE CURSOR
, LEFT AR-";
400 PRINT"ROW DISPLAYS SHAPE SET
 BY CLEAR.SHIFT CLEAR STARTS SAV
E OF ASCIIAND BINARY FILES. USE
ASCII TO  TRANSMIT, BINARY TO";
410 PRINT" EDIT WITH   THIS PROG
RAM.",,"CHARACTERS ARE TYPED AS
USUAL,  EXCEPT THAT INVERTED SYM
BOLS DO NOT PRINT."
420 PRINT"PRESS 1 TO START NEW G
RAPHIC OR        0 TO LOAD EXISTI
NG FILE.";
430 K$=INKEY$:IFK$<"0"ORK$>"1"TH
EN430ELSESOUND200,2:ER$=STRING$(
63,32):PRINT@448,ER$;
440 PRINT@448,"PRESS 1 FOR TAPE"
,"       0 FOR DISK";
450 L$=INKEY$:IFL$<"0"ORL$>"1"TH
EN450ELSEPRINT@448,ER$;:PRINT@48
0,"ENTER FILE NAME(8 CHAR. MAX.)
";:PRINT@448,"";:LINEINPUTF$
460 IFL$="1"THENDV=-1ELSEDV=1
470 IFK$="1"THENPRINT@448,"PRESS
 CLS VALUE":GOTO490ELSECLS0:IFDV
=1THENLOADMF$ELSECLOADMF$
480 FORAD=&H400 TO&H5FF:S=PEEK(A
D):GOSUB520:NEXT:SOUND200,2:RETU
RN
490 K$=INKEY$:IFK$<"0"ORK$>"8"TH
EN490ELSEIFK$="0"THENS=&H60 ELSE
S=&H70+VAL(K$)*&H10+&HF
500 FORI=&H400 TO&H5FF:POKEI,S:P
OKEI+&HA00,S:NEXT:RETURN
510 'Transfer Edit Array to Tran
smit Array
520 IF S<27THENS=S+96ELSEIFS>95A
NDS<128THENS=S-64
530 POKEAD+&HA00,S:RETURN
```

```
80.....0308
160.... 047C
END....0637
```

Listing 3

```
10 'SYSOP PROGRAM
20 CLEAR5000:DIMNAME$(50):POKE15
0,1
30 CLS:PRINT"ENTER NUMBER OF FUN
CTION:","1. DISPLAY ACTIVITY FIL
E","2. PRINT ACTIVITY FILE","3.
INITIATE ACTIVITY FILE"
40 PRINT"4. DISPLAY MEMBER FILE"
,"5. PRINT MEMBER FILE","6. STAR
T MEMBER FILE","7. REMOVE MEMBER
S","8. KILL OLD MESSAGES","9. EN
D
50 INPUTK:IFK<1ORK>9THEN30ELSEON
K GOSUB60,60,100,110,110,140,160
,210,250:GOTO30
60 IFK=1THENDV=0ELSEDV=-2
70 OPEN"D",1,"ACTIVITY":L=LOF(1)
:PRINT#DV,"CALLERS ="L
80 FORI=1TOL:GET#1,I:INPUT#1,ACT
$:FORJ=255TOOSTEP-1:IFMID$(ACT$,
J,1)=" "THENNEXTJ ELSEACT$=LEFT$
(ACT$,J)
```

```
90 PRINT#DV,ACT$:NEXTI:CLOSE:LIN
EINPUT"PRESS ENTER TO CONTINUE";
K$:RETURN
100 KILL"ACTIVITY/DAT":RETURN
110 IFK=4THENDV=0ELSEDV=-2
120 OPEN"I",1,"MEMBERS":INPUT#1,
NUM:FORI=1TONUM:INPUT#1,NAME$(I)
:PRINT#DV,LEFT$(NAME$(I),3)" "RI
GHT$(NAME$(I),LEN(NAME$(I))-3):N
EXT
130 CLOSE:LINEINPUT"PRESS ENTER
TO CONTINUE";K$:RETURN
140 CLS:PRINT"ENTER SYSOP INITIA
LS AND NAME,  FOR INSTANCE: LPLL
ANE LESTER":LINEINPUTSYSOP$
150 OPEN"O",1,"MEMBERS":WRITE#1,
1:WRITE#1,SYSOP$:CLOSE:RETURN
160 K=4:GOSUB110
170 INPUT"INITIALS OF MEMBER TO
REMOVE,    X TO RETURN";INITS$:IF
INITS$<>"X"THEN190ELSEOPEN"O",1,
"MEMBERS"
180 WRITE#1,NUM:FORI=1TONUM:WRIT
E#1,NAME$(I):NEXT:CLOSE:RETURN
190 J=1:FORI=1TONUM:IFLEFT$(NAME
$(I),3)=INITS$THENI=I+1:NUM=NUM-
1
200 NAME$(J)=NAME$(I):J=J+1:NEXT
:GOTO170
210 'Kill Old Messages
220 FORI=3TO11:DSKI$1,17,I,A$,B$
:A$=A$+LEFT$(B$,120):FORJ=0TO7:S
BJECT$=MID$(A$,J*32+1,8):EXT$=MI
D$(A$,J*32+9,3)
230 A=ASC(SBJECT$):IFA=255THENJ=
7:I=11ELSEIFA<>0THENIFMID$(EXT$,
2,1)="*"THENKILLSBJECT$+"/"+EXT$
+":1"
240 NEXTJ,I:RETURN
250 END
```

100....	039C	790....	153B
200.....	0655	900....	18A1
260....	080A	1010...	1B89
380....	0B1C	1130...	1E63
520....	0E20	1280...	216F
680.....	1277	END...	23CE

Listing 4

```
10 'RAINBOARD - Copyright 1983 b
y Lane Lester, 413 Woodland Circ
le, Lynchburg, VA 24502
20 GOTO1450
30 'Lower/Upper Case Input Routi
ne
40 PRINTMSG$;:LINEINPUTC$:LN=LEN
(C$):IFLN>1THEND$=C$:E$="":FORI=
1TOLN:C$=MID$(D$,I,1):GOSUB50:E$
=E$+C$:NEXT:C$=E$:RETURN
50 IFC$>"@"THENC$=CHR$((ASC(C$)O
R32)-32)
60 IFLN=1ANDBITS=8THENPRINTSTRIN
G$(32,RND(7)*16+143);:RETURNELSE
RETURN
70 CLEAR5000:DIMDISPLAY$(46),NAM
E$(20),TEXT$(70)
80 FILE$="WELCOME":Y=1:Z=15:GOSU
B1440:FILE$="HELP":Y=16:Z=30:GOS
UB1440:FILE$="COLDSTRT":Y=31:Z=4
6:GOSUB1440
90 CLS:LINEINPUT"IF A SINGLE-DRI
VE SYSTEM, REMOVESTARTUP DISK AN
D INSERT ONLINE  DISK. PRESS ENT
ER TO CONTINUE.";C$
100 OPEN"I",1,"MEMBERS":INPUT#1,
NUM:FORI=1TONUM:INPUT#1,NAME$(I)
:NEXT:CLOSE:CLS:PRINT"RAINBOARD
IS READY TO RECEIVE!
110 D1=0:D1$="0":D2=0:D2$="0" 'M
essage (D1) and Program (D2) Dri
ve Numbers
120 'Logon Procedure
130 LINEINPUTSTART$:FORI=1TO100:
NEXT
140 LINEINPUT"DO YOU WANT 7 BITS
 FOR TEXT ONLY OR 8 BITS FOR COL
OR, TOO? ";BITS$:IFBITS$<"7"ORBI
TS$>"8"THEN140ELSEBITS=VAL(BITS$
)
150 FL=0:IFBITS=8THENPRINTCHR$(2
7)"G4":FORI=1TO15:PRINTDISPLAY$(
I);:NEXT:GOTO170
160 PRINT"WELCOME TO THE RAINBOA
RD..."CHR$(13)CHR$(13)"WITH A RA
INBOW OF COLOR AND A"CHR$(13)"PO
T OF GOLD IN GOOD TIMES."CHR$(13
)
170 MSG$="YOUR THREE INITIALS, P
LEASE:":GOSUB40:I$=C$:ACT$="* "+
I$+" "
180 FL=FL+1:IFLEN(I$)<>3THENIFFL
=3THENNAME$="FRIEND":GOTO1280ELS
E170
190 GOSUB1410:IFMBR THENN$=NAME$
(I):S=INSTR(N$," ")-4:NAME$=MID$
(N$,4,S):INITS$=LEFT$(N$,3)ELSE2
30
200 MSG$=RIGHT$(N$,LEN(N$)-3)+",
"+CHR$(13)+"RIGHT(Y/N)? ":GOSUB4
0:IFC$<>"Y"ANDC$<>"N"THEN200
210 IFC$="Y"THENPRINT"CHECKING F
OR MESSAGES.":C=1:GOSUB650:GOTO3
60
220 'New Member Signup
230 LINEINPUT"PLEASE TELL ME YOU
R FIRST NAME: ";NAME$:LINEINPUT"
AND NOW YOUR LAST NAME:";LAST$:A
CT$=ACT$+" "+NAME$+" "+LAST$+" "
240 IFLEN(ACT$)<13THEN1280ELSEMS
G$="YOU ARE NOT YET A MEMBER OF
THE RAINBOARD."+CHR$(13)+"WOULD
```

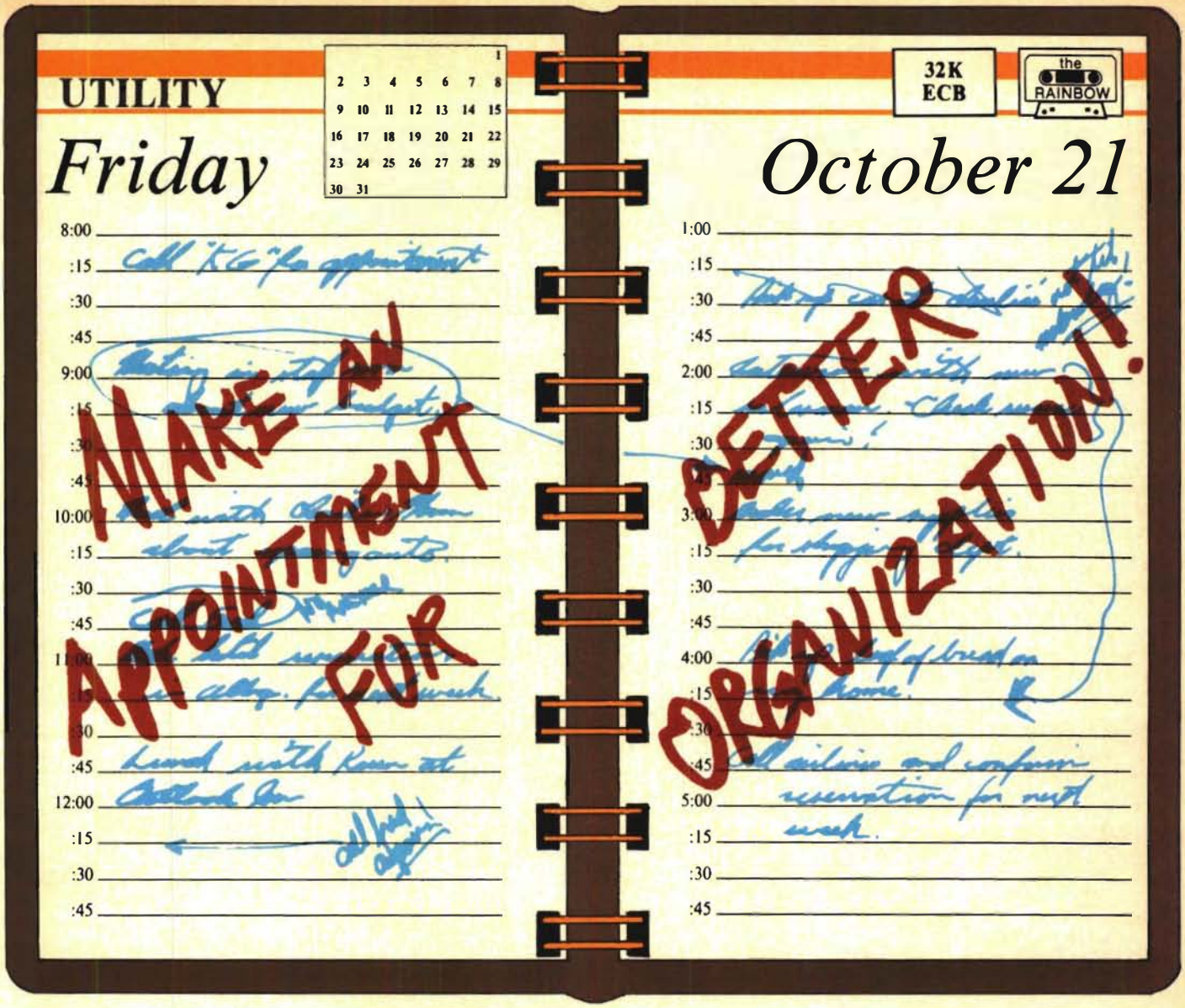
```
YOU LIKE TO BELONG? (Y/N)"
250 GOSUB40:IFLEFT$(C$,1)<>"Y"TH
EN360
260 PRINT"GREAT! TO WHAT THREE I
NITIALS"CHR$(13)"SHOULD YOUR MES
SAGES BE"CHR$(13)"ADDRESSED? ";:
LINEINPUTINITS$:FL=0
270 FL=FL+1:IFFL=6THEN1280ELSEFO
RI=1TONUM:IFLEFT$(NAME$(I),3)<>I
NITS$THENNEXT:GOTO300
280 PRINT"I'M SORRY, "NAME$", AN
OTHER"CHR$(13)"MEMBER HAS THE SA
ME INITIALS."CHR$(13)"PLEASE MAK
E UP THREE OTHERS
290 LINEINPUT"FOR THE RAINBOARD:
 ";INITS$:GOTO270
300 IFLEN(INITS$)=3THENFORI=1TO3
:A=ASC(MID$(INITS$,I)):IFA>64AND
A<91THENNEXT:GOTO330
310 ACT$=ACT$+" "+INITS$:FL=FL+1
:IFFL=6THEN1280ELSEPRINT"WE NEED
 THREE INITIALS."CHR$(13)"FOR IN
STANCE, MINE ARE 'LPL'.
320 LINEINPUT"WHAT ARE YOURS? ";
INITS$:GOTO270
330 PRINT"JUST A MOMENT WHILE I
ADD YOU"CHR$(13)"TO THE MEMBERSH
IP LIST.":NUM=NUM+1:NAME$(NUM)=I
NITS$+NAME$+" "+LAST$
```



```
340 OPEN"O",1,"MEMBERS:O":WRITE#
1,NUM:FORI=1TONUM:WRITE#1,NAME$(
I):NEXT:CLOSE:FILE$="NEWMEMBR":G
OSUB1340
350 'Main Menu
360 MSG$="ENTER H,B,S,R,L,P,T,A,
M,N,G:":GOSUB40:GOSUB1370
370 C=INSTR("HBSRLPTAMNG",C$)+1:
ONC GOSUB390,390,460,490,580,740
,950,1090,1230,1260,1280,1280:GO
TO360
380 'Help
390 IFBITS=8THENFORI=16TO30:PRIN
TDISPLAY$(I);:NEXT:RETURN
400 PRINT"(H)ELP-THIS SCREEN"CHR
$(13)"(B)ULLETIN-LATEST NEWS"CHR
$(13)"(S)CAN MESSAGE TITLES"CHR$
(13)"(R)EAD MESSAGES FROM MEMBER
S"
410 PRINT"(L)EAVE MESSAGES TO ME
MBERS"CHR$(13)"(P)ROGRAMS TO DOW
NLOAD"CHR$(13)"(T)EXT FILES TO R
EAD"
420 PRINT"(A)RT GALLERY-VIEW THE
 GRAPHICS"CHR$(13)"(M)EMBER LIST
-SEE WHO'S ON"
430 PRINT"(N)AME ENTRY-IF YOU CA
ME ON IN"CHR$(13)"   THE MIDDLE
OF THE PROGRAM"CHR$(13)"(G)OODBY
E-BEFORE HANGING UP
440 RETURN
450 'Bulletin
460 MSG$="THE CURRENT BULLETIN I
S DATED"+CHR$(13)+"9/1/83."+CHR$
(13)+"DO YOU WANT TO READ IT (Y/
N)?":GOSUB40:IFC$<>"Y"THENRETURN
470 FILE$="BULLETIN":GOSUB1340:R
ETURN
480 'Scan Messages
490 MSG$="ENTER H,C,I,R:":GOSUB4
0:GOSUB1370:C=INSTR("HCIR",C$)+1
:IFC=5THENRETURNELSEONC GOTO490,
500,510,530
500 FILE$="SCAN":GOSUB1340:GOTO4
90
510 I$="XXX":GOTO540
520 I$=INITS$:GOTO540
530 MSG$="FOR WHICH 3 INITIALS W
OULD YOU"+CHR$(13)+"LIKE TO SCAN
, "+NAME$+"? ":GOSUB40:I$=C$:IFL
EN(I$)<>3THEN530
540 CK=0:FORI=3TO11:DSKI$D1,17,I
,A$,B$:A$=A$+LEFT$(B$,120):FORJ=
0TO7:SBJECT$=MID$(A$,J*32+1,8):E
XT$=MID$(A$,J*32+9,3):A=ASC(SBJE
CT$)
550 IFA=255THENJ=7:I=11ELSEIF(EX
T$=I$ORI$="XXX")ANDEXT$>"A00"AND
EXT$<>"DAT"ANDLEFT$(EXT$,2)<>"P0
"ANDA<>0THENPRINTSBJECT$"/"EXT$:
```

```
CK=1
560 NEXTJ,I:IFCK THEN490ELSEPRIN
T"SORRY, "NAME$","CHR$(13)"NO ME
SSAGES FOUND.":GOTO490
570 'Read Messages
580 MSG$="ENTER H,S,I,R:":GOSUB4
0:GOSUB1370:C=INSTR("HSIR",C$):I
FC=4THENRETURNELSEONC+1 GOTO580,
590,610,630
590 FILE$="READ":GOSUB1340:GOTO5
80
600 GOSUB650:GOTO580
610 PRINT"WHICH SUBJECT WOULD YO
U LIKE TO READ, "NAME$"?";:LINEI
NPUTS$:L=LEN(S$):IFL>8THEN720
620 S$=S$+STRING$(8-LEN(S$)," ")
:GOSUB650:GOTO580
630 PRINT"FOR WHICH 3 INITIALS W
OULD YOU"CHR$(13)"LIKE TO READ M
ESSAGES? ";:LINEINPUTI$:IFLEN(I$
)<>3THEN630ELSEGOSUB650:GOTO580
640 'Directory Search
650 CD=0:FORK=3TO11:DSKI$D1,17,K
,AD$,BD$:AD$=AD$+LEFT$(BD$,120):
FORL=0TO7:SBJECT$=MID$(AD$,L*32+
1,8):EXT$=MID$(AD$,L*32+9,3)
660 AC=ASC(SBJECT$):IFAC=255THEN
L=7:K=11:GOTO710
670 IFAC=00R(C=1ANDEXT$<>INITS$)
OR(C=2ANDSBJECT$<>S$)OR(C=3ANDEX
T$<>I$)OR(SBJECT$="PERSONAL"ANDE
XT$<>INITS$)THEN710
680 CD=1:FILE$=SBJECT$+"/"+EXT$:
PRINTCHR$(13)FILE$:FILE$=FILE$+"
:"+D1$:GOSUB1340
690 IFEXT$<>INITS$THEN710ELSEF$=
LEFT$(FILE$,10)+"*"+RIGHT$(FILE$
,3):RENAMEFILE$TOF$
700 MSG$=NAME$+", WOULD YOU LIKE
"+CHR$(13)+"TO REPLY TO THIS (Y/
N)? ":GOSUB40:IFC$="Y"THENGOSUB7
40
710 NEXTL,K:IFCD THENRETURNELSEP
RINT"SORRY, "NAME$","CHR$(13)"NO
 MESSAGES FOUND.":RETURN
720 FILE$="READ":GOSUB1340:GOTO5
80
730 'Leave Messages
740 MSG$="ENTER 3 LETTERS:"+CHR$
(13)+"INITIALS OF RECIPIENT,"+CH
R$(13)+"MEM-BER LIST,"+CHR$(13)+
"INS-TRUCTIONS, OR"+CHR$(13)
750 MSG$=MSG$+"RET-URN TO MAIN M
ENU:":GOSUB40:I$=C$
760 IFLEN(I$)=3THENFORI=1TO3:A=A
SC(MID$(I$,I,1)):IF(A>64ANDA<91)
OR(A>47ANDA<58)THENNEXT:GOTO780
770 PRINT"WE NEED 3 LETTERS.":RE
TURN
780 IFI$="MEM"GOSUB1260:GOTO740E
LSEIFI$="INS"THENFILE$="LEAVE":G
OSUB1340:GOTO740ELSEIFI$="RET"TH
ENRETURN
790 IFRIGHT$(I$,1)>"9"THENGOSUB1
410:IFNOTMBR THENPRINT"SORRY, "N
AME$", ONLY MEMBERS"CHR$(13)"CAN
 RECEIVE MESSAGES.":GOTO740
800 PRINT"WHAT IS THE MESSAGE'S
SUBJECT?":LINEINPUT"(8 LETTERS M
AXIMUM):";S$
810 IFLEN(S$)>8THENPRINT"8 LETTE
R MAXIMUM, "NAME$:RETURN
820 IFINSTR(S$,":")ORINSTR(S$,"0
")ORINSTR(S$,"/")ORINSTR(S$,".")
THENPRINT"PLEASE DO NOT USE: 0 .
 : /":RETURN
830 PRINT"CHECKING FOR DUPLICATE
 FILE."CHR$(13)"ONE MOMENT, PLEA
SE.":FILE$=S$+STRING$(8-LEN(S$),
32)+"/"+I$+":"+D1$:CS=0
840 FORI=3TO11:DSKI$D1,17,I,A$,B
$:A$=A$+LEFT$(B$,120):FORJ=0TO7:
PRINT".";:SBJECT$=MID$(A$,J*32+1
,8):EXT$=MID$(A$,J*32+9,3)
850 IFASC(SBJECT$)=255THENJ=7:I=
11:GOTO870
860 F$=SBJECT$+"/"+EXT$+":"+D1$:
IFF$=FILE$ORF$=LEFT$(FILE$,10)+"
*"+RIGHT$(FILE$,3)THENCS=1:J=7:I
=11
870 NEXTJ,I:PRINT:IFCS THENPRINT
"SORRY, "NAME$", THAT FILE"CHR$(
13)"NAME IS ALREADY IN USE."CHR$
(13)"PLEASE  TRY AGAIN.":GOTO800
880 PRINT"BEGIN MESSAGE ENTRY."C
HR$(13)"PRESS 'ENTER' BY ITSELF
TO END.":I=-1:C$=FILE$:GOSUB1370
890 I=I+1:PRINT".":LINEINPUTTEXT
$(I):IFTEXT$(I)=""ORI=70THEN920E
LSELT=LEN(TEXT$(I)):IFLT<64THEN8
90
900 PRINTCHR$(7)CHR$(7)CHR$(7)"T
HAT LINE'S TOO LONG, "NAME$"."CH
R$(13)"IT NOW READS:"
910 FORJ=LT TO1STEP-1:IFMID$(TEX
T$(I),J,1)<>" "ORJ>65THENNEXTELS
ETEXT$(I)=LEFT$(TEXT$(I),J-1):PR
INTTEXT$(I):GOTO890
920 MSG$="ENTER 'S' TO SAVE; 'R'
 FOR"+CHR$(13)+"ANOTHER REQUEST:
 ":GOSUB40:IFC$<"R"ORC$>"S"THEN9
20
930 IFC$="R"THENRETURNELSEOPEN"O
",1,FILE$:FORJ=0TOI-1:PRINT#1,TE
XT$(J):NEXT:CLOSE:PRINT"YOUR MES
SAGE IS SAVED, "NAME$".":RETURN
940 'Programs
950 MSG$="ENTER H,L,D,P,R:":GOSU
B40:GOSUB1370:C=INSTR("HLDPR",C$
)+1:IFC=6THENRETURNELSEONC GOTO9
```

```
50,960,970,1010,1020
960 FILE$="PROGRAMS":GOSUB1340:G
OTO950
970 FORI=3TO11:DSKI$D2,17,I,A$,B
$:A$=A$+LEFT$(B$,120):FORJ=0TO7:
SBJECT$=MID$(A$,J*32+1,8):EXT$=M
ID$(A$,J*32+9,3)
980 A=ASC(SBJECT$):IFA=255THENJ=
7:I=11:GOTO1000ELSEIFA<>0ANDLEFT
$(EXT$,1)="P"ANDMID$(EXT$,2,1)="
0"THENPRINTSBJECT$" ";ELSE1000
990 T$=RIGHT$(EXT$,1):IFT$="U"TH
ENPRINT"UTILITY"ELSEIFT$="A"THEN
PRINT"APPLICATION"ELSEIFT$="G"TH
ENPRINT"GAME"ELSEPRINT"OTHER"
1000 NEXTJ,I:GOTO950
1010 FILE$="PROGDESC":GOSUB1340:
GOTO950
1020 I$=INIT$:GOSUB1410:IFNOTMB
R THENPRINT"SORRY, "NAME$", ONLY
 MEMBERS"CHR$(13)"CAN DOWNLOAD P
ROGRAMS.":RETURN
1030 PRINT"WHICH PROGRAM WOULD Y
OU LIKE TO GET?":LINEINPUTFILE$
1040 FORI=3TO11:DSKI$D2,17,I,A$,
B$:A$=A$+LEFT$(B$,120):FORJ=0TO7
:SBJECT$=MID$(A$,J*32+1,8):EXT$=
MID$(A$,J*32+9,3)
1050 IFASC(SBJECT$)=255THENJ=7:I
=11ELSEIFSBJECT$=FILE$THEN1070
1060 NEXTJ,I:PRINT"SORRY, "NAME$
","CHR$(13)"I CAN'T FIND THAT FI
LE.":GOTO950
1070 FILE$=SBJECT$+"/"+EXT$+":"+
D2$:GOSUB1340:GOTO950
1080 'Text Files
1090 MSG$="ENTER H,L,D,T,R:":GOS
UB40:GOSUB1370:C=INSTR("HLDTR",C
$)+1:IFC=6THENRETURNELSEONC GOTO
1090,1100,1110,1140,1150
1100 FILE$="TEXT":GOSUB1340:GOTO
1090
1110 FORI=3TO11:DSKI$0,17,I,A$,B
$:A$=A$+LEFT$(B$,120):FORJ=0TO7:
SBJECT$=MID$(A$,J*32+1,8):EXT$=M
ID$(A$,J*32+9,3)
1120 A=ASC(SBJECT$):IFA=255THENJ
=7:I=11ELSEIFEXT$<"AAA"ANDA<>0TH
ENPRINTSBJECT$+"/"+EXT$
1130 NEXTJ,I:GOTO1090
1140 FILE$="TEXTDESC":GOSUB1340:
GOTO1090
1150 I$=INIT$:GOSUB1410:IFNOTMB
R THENPRINT"SORRY, "NAME$", ONLY
 MEMBERS"CHR$(13)"CAN RECEIVE TE
XT FILES.":RETURN
1160 PRINT"WHICH FILE WOULD YOU
LIKE TO    READ (DO NOT INCLUDE
EXTENSION)?";:LINEINPUTFILE$
1170 FORI=3TO11:DSKI$0,17,I,A$,B
$:A$=A$+LEFT$(B$,120):FORJ=0TO7:
SBJECT$=MID$(A$,J*32+1,8):EXT$=M
ID$(A$,J*32+9,3)
1180 IFASC(SBJECT$)=255THENJ=7:I
=11ELSEIFSBJECT$=FILE$ANDEXT$<"9
99"THEN1200
1190 NEXTJ,I:PRINT"SORRY, "NAME$
","CHR$(13)"CAN'T FIND THAT FILE
.":GOTO1090
1200 FILE$=FILE$+"/"+EXT$:GOSUB1
340:GOTO1090
1210 FILE$="TEXT":GOSUB1340:GOTO
1090
1220 'Art Gallery
1230 IFBITS=7THENPRINT"SORRY, YO
U MUST RECEIVE 8 BITS"CHR$(13)"T
O VIEW GRAPHICS.":RETURN
1240 FORI=31TO45:PRINTDISPLAY$(I
);:NEXT:CLOSE:GOTO360
1250 'Member List
1260 FORI=1TONUM:PRINTLEFT$(NAME
$(I),3)" "RIGHT$(NAME$(I),LEN(NA
ME$(I))-3):NEXT:RETURN
1270 'Goodbye
1280 OPEN"D",1,"ACTIVITY":WRITE#
1,ACT$:PUT#1,LOF(1)+1:CLOSE:IFC$
="N"THEN140
1290 PRINT"GOODBYE, "NAME$"!":TI
MER=0
1300 IFTIMER<200THEN1300ELSEPRIN
T"+++";:TIMER=0
1310 IFTIMER<200THEN1310ELSEPRIN
T"ATH":TIMER=0
1320 IFTIMER<200THEN1320ELSEFL=0
:GOTO130
1330 'Display File
1340 C$=FILE$:GOSUB1370:OPEN"I",
1,FILE$
1350 IFNOTEOF(1)THENLINEINPUT#1,
TEXT$:PRINTTEXT$:GOTO1350ELSECLO
SE:RETURN
1360 'Activity Length Test
1370 LA=LEN(ACT$)+LEN(C$):IFLA>2
55THENACT$=LEFT$(ACT$,253)+" *":
GOTO1280
1380 IFLA>230THENPRINT"PLEASE FI
NISH UP SOON SO THAT"CHR$(13)"OT
HERS CAN CALL THE RAINBOARD."CHR
$(13)"THANKS, "NAME$"."
1390 ACT$=ACT$+C$+" ":RETURN
1400 'Membership Search
1410 MBR=0:FORJ=1TONUM:IFI$=LEFT
$(NAME$(J),3)THENMBR=-1:I=J:J=NU
M
1420 NEXT:RETURN
1430 'Load Graphic Screens
1440 OPEN"I",1,FILE$:FORX=Y TOZ:
INPUT#1,DISPLAY$(X):NEXT:CLOSE:R
ETURN
1450 PCLEAR1:GOTO70
```



I tend to forget dates and, subsequently, find myself in "hot water" at least once a month for forgetting a loved one's birthday, anniversary or even a lunch date. I'm lucky if I remember my *own* birthday. And yes, I bought a pocket calendar . . . but I forgot where I put it. Luckily, I found a program that solves this embarrassing problem.

Bill Bruck's *Appointment Book,* requiring 32K Extended Color BASIC and a printer, handily stores your appointments for any day within one year. It will print a calendar with any number of memos per day, though each must be 16 characters, or less.

Type in the program and *RUN* it. The first time you run the program you must create a datafile, which is a storage space on the cassette or disk where data is kept (that cannot be run). If no file is present, the program will look for one and bomb. The program will check for the presence of the disk ROM pack, and if it is there, data will be saved from the disk, otherwise, the cassette will be used.

The default for the datafile names is "DIARY." By using this name, you can press ENTER when asked for a datafile name. Occasionally, the program will not run the first time because of the *PCLEAR1* command. Stay calm and keep trying, it will work.

First, the program will ask you to input a month, day and reminder, and end month, end date. Four options are given after the item is entered:

(Y) — Put in a different item for a different day.
(N) — Return to main menu.
(S) — Put in another item for that same day.
(D) — Put in that same item again on a different day.

To delete an item, press D.

It is necessary to input the beginning month, date and day of the week (this is only crucial in the printout command), and ending month and date, so all days and dates will correspond correctly. Leap year is a special situation. As the program is presented, February 29 can exist only if you make it the beginning or end of a given display or printout. Otherwise, every fourth year will have the wrong day of the week identified for quite a few dates in succeeding months. So, to modify leap years, change line 788 by removing "VB=60 OR" and change line 2030 by removing "D=60 OR."

If you wish to move your data from cassette to disk, follow these directions:

- Put in disk controller, *CLOAD* program, *SAVE* it to disk.
- Add lines 118 DC=-1 and 1045 DC=1.
- *RUN* program, but don't save this version.

Immediately save data. Type *NEW* and run program as saved on disk. It is a good idea to save your data before exiting or printing . . . just in case of a mistake.

After entering the appropriate data, your printer will begin. At your command, the printer will provide you with data between one week to a full year in a standard calendar format. As you will see, this program utilizes your printer's capabilities by taking advantage of expanded and condensed fonts, tab controls, etc. It is suggested that you print two months at a time because the program does not skip perforations, unfortunately.

Appointment Book is written specifically for the Epson MX-80 and MX-100 printers. Since different printers use different control commands, modifications will be necessary if you do not have these printers. In this program, the following control codes are used:

Emphasized mode: CHR$(27)"E" (And cancel emphasized mode — CHR$(27)"F") in lines 680 and 700. If your printer doesn't have these, eliminate them from the program including the semicolon after them.

Double wide mode: CHR$(27)"S" — line 680 (My printer stops printing double wide at the end of the line. If yours keeps printing double wide, put the stop code at the end of line 680). If your printer does not have this, eliminate it, (including the semicolon) and adjust the space between the days of the week to get them to line up properly.

Compressed mode: CHR$(15) — line 700. This is necessary for the printout to make sense. Otherwise, you need a major rewrite of this whole section and rethinking of how many spaces are available. Substitute your code for this one.

An important note: LPVII and DMP-100 printers — This program will not run because these printers do not have compressed characters.

LPVIII and DMP-200: Use the following lines:

680 PRINT#-2, CHR$(27);CHR$(19);CHR$(27);CHR$(14);" S M (the rest as in current line 680)

700 PRINT#-2,CHR$(27);CHR$(20);CHR$(27);CHR$(15)

Bill Bruck's *Appointment Book* has been a real life—and love—saver for me. I no longer get dirty looks and the silent treatment for forgetting important dates. Now, all I need to do is find a solution to that darned name forgetting problem.

—Susan Remini

98	01A0	1210	0CC6
340	03E5	1410	0E8B
490	0582	1610	10BE
755	07C2	1810	1277
1030	0A83	END	1544

The listing:

```
10 'DIARY 3.4 2/12/83
20 GOTO 9999
30 CLEAR 10000
40 DIM I$(372)
```

```
50 DIM B$(40,2)
60 DIM M$(12)
70 DIM L$(14)
80 CL=5 'CALENDAR LENGTH
81 '     INITIALIZE
85 CLS(3):PRINT@32*8+8,"APPOINTM
ENT BOOK";
86 PRINT@32*10+8,"BRUCK ASSOCIAT
ES";
87 PRINT@32*11+8," (301) 270-582
2 ";
90 'CHECK FOR DISK BASIC
91 DC=PEEK(188)
92 IF DC=14 THEN DC=1 ELSE DC=-1
96 PRINT@32*15,"FILENAME <N> NEW
 FILE";
97 INPUT F$
98 IF F$="N" THEN INPUT"NEW FILE
 NAME";F$:GOTO 210
99   IF LEN(F$)>8 THEN PRINT"INVA
LID FILE NAME":GOTO 97
104 IF F$="" THEN F$="DIARY"
110 IF DC=-1 THEN PRINT"POSITION
 CASSETTE"
117 '     GET DATA
120 OPEN"I",#DC,F$
130 INPUT #DC,MF
140 INPUT#DC,DATE
150 FOR N=1 TO 372
160 LINE INPUT#DC,I$(N)
170 NEXT N
180 CLOSE#DC
210 CLS(3)
220 PRINT
230 PRINT"          CALANDAR PROGR
AM
240 PRINT@224,"                 INPUT
"
250 PRINT"                PRINTOUT"
260 PRINT"                DISPLAY"
270 PRINT"                SAVE"
280 PRINT@416,"      <CHOOSE ONE
PLEASE>"
290 I$=INKEY$:IF I$="" THEN 290
300 IF I$="I" OR I$="i" THEN 350
310 IF I$="D" OR I$="d" THEN 480
320 IF I$="P" OR I$="p" THEN 650
330 IF I$="S" OR I$="s" THEN 104
0
340 PRINT@448,"INVALID COMMAND":
GOTO 290
350 '     INPUT NEW DATA
360 CLS
370 GOSUB 1550
380 PRINT"What? (16 characters m
ax)"
390 LINE INPUT "?";J$
400 J=(A-1)*31+B
410 IF LEN(J$)>16 THEN 380
420 I$(J)=I$(J)+J$+"@"
430 PRINT"ANOTHER?":PRINT"<Y> YE
S":PRINT"<N> NO":PRINT"<S> SAME
DAY ANOTHER ITEM":PRINT"<D> SAME
 ITEM ANOTHER DAY"
440 I$=INKEY$:IFI$=""THEN440
450 IF I$="Y" OR I$="y" THEN 350
460 IF I$="S" OR I$="s" THEN 380
465 IF I$="D" OR I$="d" THEN GOS
UB 1550:GOTO 400
470 GOTO 210
480 '     DISPLAY
485 Z=0
490 GOSUB 1140
500 CLS
510 FOR N=DATE TO Q
520 D=N:GOSUB 1920  ' GET DATE
530 K$=I$(N)
540 IF K$="" THEN 620
550 J=INSTR(K$,"@")
560 D$=LEFT$(K$,J-1)
570 IF J=LEN(K$) THEN K$="":GOTO
    590
580 K$=RIGHT$(K$,LEN(K$)-J)
590 PRINTDT$;TAB(9)D$
600 Z=Z+1:IFZ/12=INT(Z/12) THEN
GOSUB 1480
610 IF J<>LEN(I$(N)) THEN 540
620 NEXT N
630 GOSUB 1480
640 GOTO 210
650 '          TICKLER PRINTOUT
655 EOY=0
660 D=DATE
670 GOSUB 1140
680 PRINT#-2,CHR$(27)"E";CHR$(27
)"S";"   S    M    T       W    T
    F    S";
690 PB=DATE-MF+1:PE=Q
695 IF PB>PE THEN 210
700 PRINT#-2,CHR$(27)"F";CHR$(15
)
710 PRINT#-2,STRING$(132,"-")
720 LN=0
730 '  END ROUTINE
740 IF PB>PE THEN 210
750 IF EOY=1 THEN 210
755 VB=PB
760 GOSUB 1980 'DATE ON LINE 1
770 ' BUFFER 1 - ALL WEEK ITEMS
780 FOR N=1 TO7
785 IF VB<1 THEN WR=372 :GOTO 79
0
787 IF VB>372 THEN WR=-372 ELSE
WR=0
788 IF VB=279 OR VB=124 OR VB=15
5 OR VB=341 OR VB=60 OR VB=61 OR
 VB=62 THEN VB=VB+1:GOTO 788
790 P$(N)=I$(VB+WR):VB=VB+1
```

```
800 NEXT N
810 ' BUFFER 2 - 1 LINE OF ITEMS
820 FOR N=1 TO 7
830 IF P$(N)="" THEN PL$(N)="":G
OTO 870
840 J(N)=INSTR(P$(N),"@")
850 PL$(N)=LEFT$(P$(N),J(N)-1)
860 P$(N)=RIGHT$(P$(N),LEN(P$(N)
)-J(N))
870 NEXT N
880 ' PRINT ONE LINE
890 LN=LN+1
900 FOR N=1 TO 7
910 PRINT#-2,TAB((N-1)*19)"+";PL
$(N);
920 NEXT N
930 PRINT#-2,TAB(131)"+"
940 IF LN>CL THEN 1430
950 GOTO 820
960 '   CAPITALIZE ENTRY
970 B$="":FORN=1 TO LEN(BD$)
980 A=ASC(MID$(BD$,N,1))
990 IF A>96 AND A<123 THEN A=A-3
2
1000 B$=B$+CHR$(A)
1010 NEXT N
1020 BD$=B$
1030 RETURN
1040 '     SAVE TICKLER
1042 INPUT"SAVE AS FILE";Y$:IFY$
<>"" THEN F$=Y$
1045 IFDC=-1 THEN PRINT"READY CA
SSETTE, PRESS <ENTER>":MOTOR ON
1047 IF DC=-1 THEN Y$=INKEY$:IFY
$=""THEN1047
1049 MOTOR OFF
1050 OPEN"O",#DC,F$
1060 PRINT#DC,MF
1070 PRINT#DC,DATE
1080 FORN=1TO 372
1090 PRINT#DC,I$(N)
1100 NEXT N
1110 CLOSE#DC
1120 CLS(3)
1130 PRINT"DATA SAVED":GOTO 240
1140 '    B/E DATE SUBROUTINE
1150 '    DATE=B DATE, Q=E DATE
1160 INPUT"BEGINNING MONTH <1-12
>";BM
1170 INPUT"BEGINNING DATE  <1-31
>";BD
1180 INPUT"BEGINNING DAY";BD$
1190 INPUT"ENDING MONTH     <1-12
>";EM
1200 INPUT"ENDING DATE      <1-31
>";ED
1210 IF BM<1 OR BM>12 OR BD<1 OR
 BD>31 OR EM<1 OR EM>12 OR ED<1
OR ED>31 THEN PRINT"REDO DATES":
GOTO 1160
```

```
1220 GOSUB 1770
1230 RETURN
1240 FORN=1 TO BCT
1250 IF VAL(BD$(N,2))<DATE THEN
1290
1260 IF VAL(BD$(N,2))>Q THEN 129
0
1270 M=VAL(BD$(N,2))-DATE+MF
1280 L$(M)=BD$(N,1)+" B/D@"+L$(M
)
1290 NEXT N
1300 FORN=1 TO TCT
1310 IF VAL(I$(N,1))<DATE THEN 1
350
1320 IF VAL(I$(N,1))>Q THEN 1350
1330 M=VAL(I$(N,1))-DATE+MF
1340 L$(M)=I$(N,2)+"@"+L$(M)
1350 NEXT N
1360 RETURN
1370 '     GET TICKLER ITEM FROM
STRING
1380 A=INSTR(L$(PD),"@")
1390 L$=LEFT$(L$(PD),A-1)
1400 IF A=LEN(L$(PD)) THEN L$(PD
)="" ELSE L$(PD)=RIGHT$(L$(PD),L
EN(L$(PD))-A)
1410 RETURN
1420 '    PRINTOUT OVER 6 ITEMS I
N A DAY
1430 FOR Z=1 TO 7
1440 IF P$(Z)="" THEN 1450 ELSE
     820
1450 NEXT Z
1455 PB=PB+7
1460 GOTO 710
1470 '  WAIT FOR KEY SUBROUTINE
1480 Y$=INKEY$:IFY$=""THEN1480
1490 IF Y$="@" THEN 210
1500 IF Y$="D" OR Y$="d" THEN 15
20
1510 RETURN
1520 '     DELETE ITEM SUBROUTINE
1530 GOSUB 1550
1540 GOTO 1610
1550 INPUT "DATE <MONTH>/<DATE>"
;DDT$
1560 J=INSTR(DDT$,"/")
1570 IF J=0 THEN PRINT"INVALID D
ATE":FORJ=1TO300:NEXT:GOTO 210
1580 A=VAL(LEFT$(DDT$,J-1)):B=VA
L(RIGHT$(DDT$,LEN(DDT$)-J)):D=(A
-1)*31+B
1590 IF A<1 OR A>12 OR B<1 OR B>
31 THEN PRINT"INVALID DATE":FORJ
=1 TO 300:NEXT:GOTO 210
1600 RETURN
1610 IF INSTR(I$(D),"@")=LEN(I$(
D)) THEN 1620 ELSE 1630
1620 I$(D)="":PRINT"ITEM DELETED
":GOTO 500
1630 PRINT@481,"DELETE NUMBER";
1640 INPUT X
1650 IF X=1 THEN 1740
1660 FOR N=1 TOX
1670 J=INSTR(J+1,I$(D),"@")
1680 IF J=0 THEN PRINT"INVALID N
UMBER":FORJ=1TO300:NEXT:GOTO 500
1690 IF N=X-1 THEN A=J
1700 NEXT
1710 I$(D)=LEFT$(I$(D),A)+RIGHT$
(I$(D),LEN(I$(D))-J)
1730 GOTO 500
1740 J=INSTR(I$(D),"@")
1750 I$(D)=RIGHT$(I$(D),LEN(I$(D
))-J)
1760 GOTO 500
1770 '    CONVERT BEGIN/END DATES
1780 DATE=31*(BM-1)+BD
1790 Q=31*(EM-1)+ED
1800 IF Q<=DATE THEN PRINT"INVAL
ID DATES":GOTO 1160
1810 GOSUB 960
1820 C$=LEFT$(BD$,2)
1830 IF C$="SU" THEN MF=1
1840 IF C$="MO" THEN MF=2
1850 IF C$="TU" THEN MF=3
1860 IF C$="WE" THEN MF=4
1870 IF C$="TH" THEN MF=5
1880 IF C$="FR" THEN MF=6
1890 IF C$="SA" THEN MF=7
1900 IF MF=0 THEN PRINT"INVALID
DAY":GOTO 1180
1910 RETURN
1920 '    CONVERT DATE SUBROUTINE
1930 IF INT(D/31)=D/31 THEN MONT
H=INT(D/31):DY=31:GOTO 1960
1940 MONTH=INT(D/31)+1
1950 DY=D-(MONTH-1)*31
1960 DT$=STR$(MONTH)+"/"+RIGHT$(
STR$(DY),LEN(STR$(DY))-1)
1970 RETURN
1980 ' PRINT DATE ON FIRST LINE
1990 FOR N=1 TO 7
2000 D=PB+N-1
2005 IF D>372 THEN D=D-372
2010 IF D<0 THEN D=D+372
2015 IF D=0 THEN D=372
2020 IF D=279 OR D=124 OR D=155
OR D=341 THEN PB=PB+1:GOTO 2000
2030 IF D=60 OR D=61 OR D=62 THE
N PB=PB+1:GOTO 2000
2040 GOSUB 1920
2050 PRINT#-2,TAB((N-1)*19)"+";D
T$;
2060 NEXT N
2070 PRINT#-2,TAB(131)"+"
2080 RETURN
2999 '  NEW DATA FILE SUB
9998 PCLEAR 1
9999 GOTO 30
```

16K DISK

Master Messenger

By Alan Davenport

Master Messenger is a program for network addicts, like myself. It allows the user to write messages for bulletin boards off line and reduce connect time, phone bills and most important, save money. *Master Messenger* was specifically designed to be compatible with CompuServe's Special Interest Group (SIG) message boards. It also can be used to write letters to be sent by electronic mail (EMAIL) on the CompuServe system. Using this program, a network addict no longer need be afraid to spend too much time and money answering messages. Answers can be typed at leisure off line and uploaded anytime. The only danger is to the person that you are writing to—the danger of having to read a long message! Your answers to messages will be better, less hurried, and more of a joy because you will no longer be rushing to answer with one eye on the all-terrible clock.

Master Messenger is user-friendly. There are error traps so that you will not have to worry about typing a long message and then lose it.

The program has these features:

Allows typing of uploadable text compatible with smart terminal programs.
Automatic line numbering of message.
Listing of the text to the screen or to your printer.
Edit a line by replacing an error in it rather than retyping the whole line.
Insert lines in the middle of the body of the text.
Storing of text to disk or loading of an old file from disk to be re-sent, finished, or edited.
List your disk's directory on your printer.

Program Use

Master Messenger has been designed for ease of use; it has been thoroughly tested in actual use for several months and has been steadily revised. This is the 10th version. Although it is easy to use, it is important that you read these instructions. When you first type *RUN*, you will get a syntax error. Just type *RUN* again. This is not a bug in the program, it is caused by Extended Color BASIC's handling of the *PCLEAR* command. After typing *RUN* the second time, you will be greeted by the main menu. To start entering text, select [1]. Just type until you have typed about two lines worth of text onto the screen then press [ENTER]. Your line is now stored

in memory. If you type more than two lines, you may exceed the maximum line length of 79 characters (two lines is 64 characters) allowable by CompuServe's SIG message boards. If you exceed the maximum length, you will be informed, the line will not be entered into memory, and the line will be listed so that you can break it into smaller pieces. Continue entering text this way until done.

At the bottom of the screen is a display showing how many characters long your message is. CompuServe documentation says that you can enter a message up to 2,500 characters long. It is actually only possible to store a message about 2K (2048) characters long. There will be an audible and printed warning when you are have only 200 characters left. If you excede the maximum allowable message length, *Master Mesenger* will not store the line and you will be given the choice of editing the message or saving the first part and finishing your message as a separate file and uploading it as a separate message.

At any time, you can return to the main menu by typing three Ms [MMM] on a blank line or at the end of a line of text. If you do, however, that line will not be entered into memory. You can also delete the current line being typed by entering three Ds [DDD] at the end of the line.

Editing

When you are finished typing your message, you can then list, print or edit it. Select [2] from the main menu. You will then see another menu. Pressing [1] on the new menu will list the message to the screen. If, when you list your text, you see a line with an error, note the line number so you can edit it later. Press [2] to print the message. You will be asked if your printer is ready. If your printer is not ready press [N], otherwise press [Y]. To edit, press [3]. When you select editing, you will be asked for a line number. If you just hit [ENTER], *Master Messenger* will start at the first line. If you continue hitting [ENTER], the next lines will come up in sequence one at a time. If you decide to edit a line, enter a replacement string. Here is how it works. Those familiar with CompuServe's SIG editor will find that *Master Messenger* works in the same way; say that you mistyped the word "THE" as "HTE." When asked for the "string to replace," type "HTE." You will then be asked for a replacement string. Simply type "THE" correctly and hit [ENTER]. The line will then be fixed, printed and you will be asked if it is OK. If it is *not* OK press [N] and the line will not be changed. You will then be given the chance to correct the line again. When done press [Y] and the next line will appear. At any time, you can enter three other functions when asked for the "string to replace." As mentioned before, entering [MMM] will return you to the main menu. [BBB] will move you back a line. Entering [DDD] will cause the current line to be deleted from the memory.

Other Functions

From the main menu you can also perform other functions. To load a file from disk, select [3]. Just enter a valid filename and it will be loaded into memory. The file must be stored in the ASCII format. Any old message will be scratched (erased) and the new file from the disk will take its place. You can clear memory by selecting menu item number [4]. To store your text, press [5]. You will be asked for a filename. Enter any valid filename with extention and disk drive number (FILENAME.EXT:1). If you type a filename that is too long, you will be asked for another.

Master Messenger also checks to see if the disk that you are trying to store your text on is full. If it is, you will be asked to substitute another. When you have inserted a different disk, just press [ENTER] and your text will be stored or enter an [M] to return to the menu. Entering [6] from the menu will end the program. You will be asked if you really want to (so you do not accidentally lose your message!). Press [E] and *Master Messenger* will end, delete itself from memory, and reset BASIC. Use care here! When you first type it in, *SAVE* a copy before trying this or you will have to do it again! Selecting menu item [7] will allow you to insert a line in the middle of your message. You will be asked for a line number. Enter the number of the line that you want to insert the new line *before*. If your text is long, it may take a few seconds to open up a space for your new line. Menu item [8] will list the directory of any of your disks on your printer. It is useful to have a copy of your disk's directory handy when on-line, especially if you are sending multiple messages and cannot remember a filename!

I have been using this program without problems for several months now using Eigen System's *Colorcom/E* smart terminal program. To send a message using *Colorcom/E* select [R] from the menu. To leave a message on a CompServe SIG, select transmit option 1, "NONE." Enter [L] at the SIG's function prompt, then enter who you want to send your message to. When you see the first line number, (1:), press the down arrow on your keyboard and the number [2]. Your message will then be uploaded (transmitted) to the SIG message board. It will appear that the message is not being received properly, line numbers will appear in odd places, but when the message is completely uploaded, press [ENTER], type [P], then press [ENTER] again and you will see that your message is stored on the board. After you see that it is okay, type [S] and a section number (SO) and you will receive the message "Message #12345 stored."

To send EMAIL (electronic mail), first load your text into *Colorcom/E* by selecting transmit option [2], "ADD HEADER." Then to go to CompuServe's EMAIL section, type [GO EMA]. On the menu that you see, press [22], "To send mail." On the next menu, select item [1], "FILGE Editor." When you see "ready." press ↓ and the number [2] and your message will be sent. When it is all sent, type [/EX] and you will return to the menu. Select [6] and answer the questions as to who you want to send your message to.

For other terminal programs, refer to their instructions for how to upload a file.

There are many error traps built in *Master Messenger*; however, it may still be possible to crash the program. (I/O ERROR on a disk, etc.) Many times, it is possible to recover from an error by typing *GOTO* 480. (The start of the main manu.) In many cases, your message will still be there.

That is it! This program has saved me hundreds of dollars in connect time. I hope that you find it as useful as I have. If you have questions, you can reach me at CompuServe Number [72215,743] or write me at 31 Madison Drive, Ogdensburg, N.J., 07439.

The listing:

```
130...01BC
350...0598
480...08FF
690...0C0A
980...100E
1170..12AC
1360..1554
END...17E9
```

```
10 'MASTER MESSENGER #10-7/20/83
20 'BY ALAN DAVENPORT
30 '31 MADISON DRIVE
40 'OGDENSBURG, NJ 07439
```

```
50 'COMPUSERVE # 72215,743
60 PCLEAR1:CLEAR2500:DIMA$(150):
GOTO480
70 '**************************
80 CLS
90 IFA=0THENPRINT@192,"THERE IS
NO MESSAGE TO EDIT!!!":PLAY"T1CP
1":RETURN
100 PRINT"ENTER "999" TO RETURN
TO MENU.  PRESS <ENTER> FOR LINE
 #1.":IF EL=998 THEN EL=0:RETURN
110 INPUT"EDIT WHICH LINE";EL
120 EL=EL-1:IFEL>A THEN80 ELSE I
F EL<0 THEN EL=0
130 CLS:PRINT"LINE #"EL+1:PRINTA
$(EL):PRINT
140 PRINT"ENTER <DDD> TO DELETE
THIS LINE.ENTER <MMM> TO RETURN
TO MENU.  ENTER <BBB> TO GO BACK
 A LINE.  PRESS <ENTER> TO SKIP
THIS LINE."
150 PRINT"STRING TO REPLACE?":LI
NEINPUT"";RP$
160 IF RP$="DDD" THENPRINT"ARE Y
OU SURE YOU WANT TO DELETE THIS
LINE? PRESS <Y> OR <N>."ELSE180
170 Z$=INKEY$:IFZ$="Y"THEN180ELS
EIF Z$="N"THEN130ELSE170
180 IFRP$=""THENEL=EL+1:GOTO340
190 IF RP$="MMM" THENRETURN ELSE
 IF RP$="DDD"THEN CC=CC-LEN(A$(E
L)):CLS:NL$="":PRINT@196,"<<<<<<
LINE DELETED>>>>>>":PLAY"T2AP2":
GOTO320ELSE IF RP$="BBB"THENEL=E
L-1:GOTO340
200 FORR=1TO LEN(A$(EL))-LEN(RP$
)+1
210 IF RP$=MID$(A$(EL),R,LEN(RP$
)) THEN 240
220 NEXT
230 CLS:PRINT@96,"STRING TO REPL
ACE NOT FOUND!":PLAY"T22CDCDT1P1
":GOTO130
240 CC=CC-LEN(RP$)
250 L$=LEFT$(A$(EL),R-1)
260 R$=RIGHT$(A$(EL),LEN(A$(EL))
-R-LEN(RP$)+1)
270 PRINT:PRINT"REPLACEMENT STRI
NG?":LINEINPUT"";RP$
280 NL$=L$+RP$+R$:CC=CC+LEN(RP$)
290 CLS
300 PRINT:PRINTNL$:PRINT:PRINT"I
S THIS OK? PRESS <Y> OR <N>."
310 Z$=INKEY$:IF Z$="Y"THEN 320
ELSE IF Z$="N" THEN 130 ELSE310
320 CLS
330 A$(EL)=NL$:EL=EL+1
340 IFEL<0THENEL=0 ELSE IFEL>=A
THEN CLS:PRINT@129,"<<END OF FIL
E. DONE EDITING?>>          PRESS
 <Y> OR <N>.":GOTO360
350 GOTO130
360 Z$=INKEY$:IF Z$="Y"THEN RETU
RN ELSE IFZ$="N" THENEL=0:GOTO13
0 ELSE360
370 '**************************
380 A$="":PRINT@0,"ENTER <MMM> F
OR MENU OR","ENTER <DDD> AT THE
END OF A     LINE TO DELETE THAT
 LINE.","PRESS enter TO STORE TH
E LINE."
390 PRINT@485,"CHARACTER COUNT="
CC".";:IF CC>1848 THENPRINT@416,
"*WARNING*! RUNNING OUT OF ROOM.
          FINISH MESSAGE!";:PLAY"
T2C"
400 PRINT@160,"":LINEINPUT"";A$
410 IF RIGHT$(A$,3)="DDD"THEN A$
="":SOUND1,6:PRINT@192,STRING$(2
55," ")
420 IF A$="" THEN400
430 IF RIGHT$(A$,3)="MMM"THEN480
440 IF LEN(A$)>78 THEN CLS:PRINT
@320,"LINE TOO LONG! 79 CHARACTE
RS    MAXIMUM. DO AGAIN.",A$:SOU
ND1,9:GOTO380
450 CC=CC+LEN(A$):IF CC>2048 THE
N PRINT@193,"LINE NOT ENTERED. M
ESSAGE TOO   LONG FOR MESSAGE BO
```

```
ARD! ENTER   <MMM> FOR MENU. EDI
T MESSAGE OR CONTINUE WITH A NEW
 FILENAME.   STORE NEW FILE AS A
 SEPERATE    MESSAGE ON THE BOAR
D!!":SOUND1,19:GOTO400
460 A$(A)=A$:A=A+1:SOUND200,1:A$
="":PRINT@192,STRING$(255," "):P
RINT:GOTO390
470 '************************
480 CLS
490 PRINT"<<<<<<<<<<<MAIN MENU>
>>>>>>>>>>"
500 PRINT"1. CREATE TEXT OR CONT
INUE."
510 PRINT"2. LIST OR EDIT MESSAG
E."
520 PRINT"3. LOAD FILE FROM DISK
."
530 PRINT"4. SCRATCH MESSAGE OR
OLD FILE."
540 PRINT"5. STORE MESSAGE TO DI
SK."
550 PRINT"6. EXIT PROGRAM."
560 PRINT"7. INSERT A LINE BETWE
EN OTHERS.";
570 PRINT"8. PRINT DISK DIRECTOR
Y.
580 PRINT:PRINT"<<press number o
f your choice.>>"
590 PRINT"<<<<<<<<MASTER MESSENG
ER>>>>>>>>       BY ALAN DAVENPO
RT"
600 CH=VAL(INKEY$):IF CH=0 OR CH
>8 THEN600
610 CLS
620 ON CH GOSUB 380,1260,1450,65
0,700,860,980,1170
630 GOTO480
640 '************************
650 CLS
660 PRINT"ARE YOU SURE YOU WANT
TO SCRATCHYOUR MESSAGE?":PRINT:P
RINT"          PRESS <Y> OR <N>.
670 Z$=INKEY$:IF Z$="Y" THEN 680
 ELSE IF Z$="N" THEN RETURN ELSE
 670
680 FORR=0TOA:A$(R)="":NEXT:CC=0
:A=0:PRINT@229,"OLD MESSAGE SCRA
TCHED";:PLAY"T1L2C":RETURN
690 '************************
700 CLS:IF A$(0)="" AND A=0 THEN
 PRINT@230,"NO MESSAGE TO STORE!
":SOUND1,19:RETURN
710 PRINT@192,"PRESS enter TO AB
ORT OPERATION.      ENTER <FILE
NAME/EXT:0>."
720 LINEINPUT"OUTPUT FILE NAME?
";NM$
```

```
730 IF NM$="" THEN RETURN
740 IF MID$(NM$,LEN(NM$)-1,1)=":
"THEN D=VAL(RIGHT$(NM$,1))
750 DF=FREE(D):IF DF<1 THENCLS:S
OUND1,19:PRINT:PRINT"DISK IS FUL
L!! SWITCH DISKS,     PRESS ENTER
 TO STORE. ENTER <m> TO RETURM T
O MENU.":LINEINPUT";Z$:IF Z$="M
"THEN RETURNELSE750
760 IF LEN(NM$)>14 THEN CLS:PRIN
T"        INVALID FILENAME!":SOUN
D1,1:GOTO710
770 OPEN "O",#1,NM$
780 FORR=0 TO A
790 IFA$(R)="" THEN810
800 PRINT#1,A$(R)
810 NEXT
820 CLOSE
830 PRINT:PRINTNM$" SAVED TO DIS
K."
840 PLAY"T22CFA":FORD=1TO2000:NE
XT
850 RETURN
860 CLS:PRINT:PRINT:PRINT"PRESS
<E> TO END OR ANY OTHER   KEY TO
 RETURN TO MAIN MENU."
870 Z$=INKEY$:IFZ$=""THEN870 ELS
E IF Z$<>"E" THEN 480 ELSE CLS
880 PRINT@224,"<<<<<<<<<<<END OF
 RUN>>>>>>>>>>>":PLAY"T1P1":CLS:
EXEC&H8000
890 '**************************
900 CLS
910 FORR=0TOA-1
920 PRINTR+1": "A$(R)
930 SF=SF+LEN(A$(R)):IF SF<200TH
EN 950
940 SF=0:PRINT@484,"PRESS SPACEB
AR FOR MORE";:IFINKEY$<>" "THEN9
40 ELSE CLS
950 NEXT
960 PRINT@482,"PRESS SPACEBAR TO
 CONTINUE.";:IFINKEY$=" "THEN SF
=0:A$="":CLS:RETURN ELSE 960
970 '**************************
980 CLS
990 IF A$(0)="" AND A=0 THENPRIN
T"THERE ARE NO LINES IN MEMORY!"
:SOUND1,9:FORD=1TO999:NEXT:RETUR
N
1000 INPUT"INSERT NEW LINE BEFOR
E WHICH    LINE. HIT ENTER TO AB
ORT INSERT.";NL
1010 IF NL>A THENPRINT"LINE NUMB
ER OUT OF RANGE!":PLAY"T2C":GOTO
1000
1020 PRINT@224,"STAND BY, OPENIN
G SPACE FOR NEW LINE."
1030 IF NL=0 THEN RETURN
1040 A=A+1
```

```
1050 FORR=A TO NL STEP-1
1060 A$(R)=A$(R-1):A$(R-1)=""
1070 NEXT
1080 CLS
1090 PRINT"      CHARACTER COUNT=
"CC"."
1100 PRINT@192,"LINE TO INSERT:"
1110 LINEINPUT"";NL$
1120 IF LEN(NL$)>78 THEN CLS:PRI
NT"LINE TOO LONG. 79 CHARACTERS
   MAXIMUM! DO AGAIN.":SOUND1,9:
PRINT@320,NL$:GOTO1100
1130 CLS:PRINTNL$
1140 PRINT:PRINT"IS THIS OK? PRE
SS <Y> OR <N>."
1150 Z$=INKEY$:IF Z$="Y" THEN A$
(R)=NL$:CC=CC+LEN(NL$):RETURN EL
SE IF Z$="N" THEN1080 ELSE1150
1160 '**************************
1170 CLS
1180 PRINT"LISTING DIRECTORY TO
PRINTER.   ENTER <9> TO RETURN T
O MENU."
1190 PRINT@192:INPUT"WHICH DRIVE
";DR
1200 IF DR=9 THEN RETURN
1210 REM'CHANGE "ND" IN LINE $$
1220 TO EQUAL THE HIGHEST DRIVE
$$$$ NUMBER IN YOUR SYSTEM.$$$$$
```



```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
1220 ND=1:IF DR<0 OR DR>ND THEN
PRINT@0,"<<<<<INVALID DEVICE NUM
BER>>>>>":SOUND1,5:GOTO1190
1230 POKE111,254:DIRDR
1240 PRINT#-2:RETURN
1250 '**************************
1260 CLS
1270 PRINT:PRINT"<<<<<<<<MASTER
MESSENGER>>>>>>>>":PRINTSTRING$(
32,"*")
1280 PRINT"1. LIST MESSAGE TO SC
REEN."
1290 PRINT"2. LIST MESSAGE TO PR
INTER."
1300 PRINT"3. EDIT MESSAGE."
1310 PRINT"4. RETURN TO MAIN MEN
U."
1320 PRINT:PRINTSTRING$(32,"*")
1330 PRINT"<<press number of you
r choice>>"
1340 CH=VAL(INKEY$):IF CH>4 OR C
H=0 THEN 1340
1350 ON CH GOSUB 900,1380,80
1360 RETURN
1370 '**************************
1380 CLS
1390 PRINT@224,"PRINTER READY? P
RESS <Y> OR <N>."
1400 Z$=INKEY$:IF Z$="Y" THEN 14
10 ELSE IF Z$="N" THEN RETURN EL
SE 1400
1410 FORR=0TOA-1
1420 PRINT#-2,R+1": "A$(R)
1430 NEXT:RETURN
1440 '**************************
1450 CLS
1460 PRINT"  INPUTTING OLD FILE
FROM DISK",STRING$(32,239)
1470 PRINT"ENTER <D> FOR DISK DI
RECTORY.   ENTER <M> TO RETURN T
O MENU OR  TYPE <FILENAME.EXT:0>
1480 LINEINPUT"INPUT FILE NAME?"
;NM$
1490 IF NM$=""THEN1450 ELSE IF N
M$="M"THEN RETURN ELSE IF NM$="D
" THEN DIR:GOTO1480
1500 IF LEN(NM$)>14 THEN CLS:PRI
NT@232,"FILENAME INVALID":SOUND1
,19:GOTO1480
1510 CLS4:GOSUB680
1520 OPEN "I",#1,NM$
1530 IF EOF(1)=-1 THEN 1560
1540 LINEINPUT#1,A$(A)
1550 CC=CC+LEN(A$(A)):A=A+1:GOTO
1530
1560 CLOSE:CLS
1570 PRINT@224,NM$" LOADED FROM
DISK.":PLAY"T22CBAT1P1":RETURN
```

An Automatic Phone Dialer For Radio Shack's Modem II

By Jorge Mir

A few months ago, I decided to "byte" the bullet and purchase a Radio Shack Modem II to replace my old one. The ability to dial a phone number from the keyboard, as well as the auto-answer capability of such a modem, attracted me to my new acquisition.

I am constantly using several favorite bulletin board services (some are long distance calls, such as Connection-80) as well as CompuServe and The Source. Keeping track of all these different phone numbers while having a computer at my fingertips did not make any sense at all. So, *Telecom* was "born" to make things easier for me.

Telecom will automatically dial any telephone number included in the data statements. It will identify it as a local or long distance phone number and it will determine if it is a voice or data phone call. If it is a data type phone call, it will load your terminal program and *EXECute* it after which you can press *BREAK* and type in an "X" to activate the modem for dialing the phone number. If it is a voice type phone call, the modem will dial the phone and ask you to pick up the handset after it starts ringing.

The program has sufficient prompting throughout its various subroutines, so detailed instructions are not necessary. Also, I have added sufficient REM statements to explain what each section of the program is to accomplish.

Please note that the phone numbers are entered as data statements which contain the name of the person or service, the phone number and an identification code to determine whether the number is a computer service (a number 1), or a phone number for voice communications (a number 2).

Step 650 checks to see if the number is longer than seven digits. If it's longer than seven digits, it recognizes it as a long distance phone number and adds additional digits for dialing through phone services such as MCI. In this case, I have used the phone number 222-3333 and account number 12345 for illustration purposes and you will have to change the program step to your own number and account. Note the two "ps" added for pausing approximately four seconds to wait for the tone on these services before you can enter the account code followed by the phone number to be called. If you are not using this type of service, just replace these numbers with a "1" to enter the direct dialing long distance network.

A few data statements have been added to illustrate the format required.

Please note that step 290 loads a disk version of *Videotex*. You will have to change this step according to the type of terminal program you have.

As a final point, if the telephone number you are searching for is not included in the data statements, you will have a chance to enter it from the keyboard. Of course, you can also stop the program at that point and add it to the data statements after which you should save the program so that it would contain that phone number next time you call for it.

Although the program is designed for a Radio Shack Modem II, it can be changed to work with other types of programmable modems.

```
40..... 013E
280.... 03F9
520.....0758
END... 0A35
```

The listing:

```
0 '*****************************
1 ' TELECOM -- R.S. MODEM II
2 '  AUTOMATIC PHONE DIALER
3 '       BY:  JORGE MIR
4 '*****************************
10 GOSUB610 ' print heading
20 PRINT @448,STRING$(32,140);:P
RINT@480,"   PRESS enter TO END
PROGRAM";
30 PRINT@164,"WHO DO YOU WISH TO
 CALL?"
40 PRINTTAB(8);:LINEINPUTID$:IF
ID$=""THEN CLS:END
50 '*****************************
60 '   FIND TELEPHONE NUMBER
70 '*****************************
80 RESTORE
90 READ NAME$:IF NAME$="END" THE
N 390
100 READ PHONE$,TYPE
110 IF INSTR(NAME$,ID$)=0 THEN 9
0
120 GOSUB 610:PRINT@164,"CALL "N
AME$"?":SOUND100,1
130 I$=INKEY$:IF I$=""THEN130
140 IF I$="N" THEN 90
150 '*****************************
160 '   PROGRAM R.S. MODEM II
```

(Jorge Mir, a Certified Public Accountant, is currently controller of a Fortune 500 Corporation. Personal computing is his main hobby and he publishes most of his original work through the Rainbow.)

```
170 '*****************************
180 GOSUB610:GOSUB650
190 PRINT@165,"PROGRAMMING MODEM
 FOR"
200 IF RIGHT$(NAME$,1)="S"THEN N
AME$=NAME$+"'" ELSE NAME$=NAME$+
"'S"
210 PRINT TAB((32-LEN(NAME$))/2)
NAME$
220 PRINT TAB(10)"PHONE NUMBER"
230 GOSUB540
240 IF TYPE=2 THEN 330
250 '*****************************
260 '  LOAD TERMINAL PROGRAM
270 '*****************************
280 PRINT@480,"LOADING COMMUNICA
TIONS PROGRAM";
290 !M"VIDEOTEX":POKE&HFF40,0:EX
EC
300 '*****************************
310 '  DIAL PHONE NUMBER (VOICE)
320 '*****************************
330 PRINT:PRINT"PHONE NUMBER IS
NOW BEING DIALED":PRINT#-2,"X";
340 PRINT@480,"(GET PHONE AND DI
SCONNECT MODEM";:POKE&H5FF,105
350 FOR D=1TO2000:NEXTD:CLS:END
360 '*****************************
370 'ENTER NUMBER FROM KEYBOARD
380 '*****************************
390 GOSUB610
400 PRINT@161,"SORRY, I DON'T RE
COGNIZE THAT   NAME."
410 PRINT:PRINT" DO YOU WANT TO
INPUT TELEPHONE  NUMBER FROM THE
 KEYBOARD FOR":PRINT" ";ID$" (Y/
N)?"
420 I$=INKEY$:IF I$=""THEN420
430 IF I$="N" THEN 10
440 IF I$<>"Y" THEN SOUND100,2:G
OTO420
450 PRINT:INPUT" PHONE NUMBER";N
$
452 PHONE$="":FOR N=1 TO LEN(N$)
454 X$=MID$(N$,N,1):IF INSTR(" -
()",X$)>0 THEN 458
456 PHONE$=PHONE$+X$
458 NEXT N
460 INPUT" TYPE (1=BBS, 2=VOICE)
";TYPE
470 IF TYPE<1 OR TYPE>2THEN460
480 NAME$=ID$
490 GOTO 180
500 END
510 '*****************************
520 '  SEND DIGITS TO MODEM II
530 '*****************************
540 POKE&H96,&HBE
550 FOR X=1TOLEN(N$)
560 I$=MID$(N$,X,1):IF INSTR(" -
()",I$)>0 THEN 600
570 PRINT@367,MID$(N$,X,1);:SOUN
D150,1
580 PRINT#-2,MID$(N$,X,1);
590 FOR D=1TO200:NEXTD
600 NEXTX:PRINT@367," ";:RETURN
610 CLS:PRINT TAB(5)"AUTOMATIC P
HONE DIALER":PRINT STRING$(32,13
1);:RETURN
620 '*****************************
630 '  LONG DISTANCE OR LOCAL?
640 '*****************************
650 IF LEN(PHONE$)>7 THEN N$="*D
T2223333PP12345"+PHONE$:ELSE N$=
"*DT"+PHONE$
660 RETURN
670 '***** PHONE NUMBERS ******
680 'DATA: NAME, PHONE, TYPE
690 '(ENTER NO SPACES OR DASHES)
700 'TYPE: 1=DATA, 2=VOICE
710 '*****************************
720 DATA COMPUSERVE,4756935,1
730 DATA CONNECTION 80,212441375
5,1
740 DATA SOURCE,7851614,1
750 DATA RAINBOW,5022284492,2
760 DATA MOM AND DAD,1112223333,
2
770 DATA END
```

The Shell Game

By Dale L. Puckett
Rainbow Contributing Editor

Welcome to KISSable OS-9, Month Two. In our first column we talked about OS-9 in general and tried to get you excited about the prospect of being able to do more than one thing at a time — or even let more than one person work on your CoCo at the same time.

We told you that OS-9 works because it is made up of a number of small modules that can be tailored to fit any hardware you want to run it on and introduced you to a few modules which are common to all versions of OS-9. This month we'll move on to take a look at the SHELL — the heart of OS-9 operation.

An Invitation

As the newly elected president of the OS-9 User's Group, I would like to invite you and your local CoCo club to join us. One of the goals of our group is education. We want to help others learn OS-9. By the time you read this our officers will have held their first organizational meeting on CompuServe. We'll pass along the details and our new address as soon as we have it. We're hoping that Color Computer clubs around the country will become the heart of our OS-9 users group.

What Do You Want To Know About OS-9?

As we mentioned during our first column last month, this is your column. We will attempt to give you the coverage you want about CoCo's new OS-9 operating system. We'll try very hard to find answers to your technical questions about OS-9. I plan to reserve a portion of the column each month for these answers. This is where you come in. If you have questions — or are just a little confused about this complex operating system — jot them down and mail them to me at *Rainbow*. We'll be using your letters to keep the column on track.

Again, Why OS-9

I looked back over last month's column and noticed something which told me I should give you a recap. Last month, we only gave you technical reasons why you should learn and use OS-9. I sort of got the cart before the horse. Let's try again.

The most important reason is the fact that OS-9 is on its way to becoming the standard operating system for 6809 based micros. I didn't realize how many people were using OS-9 until I attended the seminar in Des Moines. More than 50 different microcomputers run OS-9 now. These computers are alive and well on the SS-50 buss, the Motorola Exorbus, the STD buss and yes, even the S-100 buss. OS-9 is also running on several of Europe's major computers and several systems in Japan.

Now that Tandy has made it the Color Computer operating system, there will be tens of thousands of OS-9 users in the United States within a year. Also, the Tano Corporation, who bought the licensed OS-9 and plans to make it the standard operating system on those computers.

If you're looking for more hardware reasons to consider OS-9, think about the future. What happens when you fall in love with OS-9 and decide to put everyone in the office on your system. While there's not enough memory in your Color Computer to do the job, you won't need to look far to fill your expansion itch. For example, the GIMIX Level III machines access a million bytes of memory and run a half dozen terminals at the same time and do it very effectively. The direct memory access disk controllers and intelligent input/output cards make it all possible.

Yet, hardware that runs OS-9 is only half the story. How about software? Why should you buy OS-9? It's new so there's probably no software support for it, right?

Wrong! Just about every major language and other type of system software that is available on the IBM and clones, and the Apple, et al, is running today on OS-9 computers. And, most of these programs run faster on our 6809s. The applications software you need to run your business is running right now, too. And most of it is superior to similar programs running on those Z-80 and 8088 machines.

DynaCalc for example, is superior to *VisiCalc*. *DynaStar* does everything *WordStar* does — besides, it's easier to use. OK, I realise not everyone likes *WordStar* . In that case, try *Stylo*. It also beats the pants off anything the IBM clan can offer. All the utilities you need are running right now too. There's *DynaSpell, DynaMail* and *Dynamite* to name just three. Would you believe they all come from different companies? And don't forget: Computerware's *Inventory Control* and *Order Entry System*, or the RMS record management system to name just a few applications packages.

(Dale L. Puckett is a freelance writer and programmer who has worked with the Motorola family of microprocessors since 1976. He just completed his first book, "A Complete Tour Guide to BASIC09, *"this summer. It is being published by Microware and will be available this fall. He is the author of DynaSpell, Readtest, Esther and Help, which are available from Frank Hogg Laboratories. He serves on the Info World Software Review Board and is a Chief Warrant Officer in the U.S. Coast Guard.)*

If you don't believe there's a lot of software out there for OS-9, ask Dan Downard, our Technical Editor at *Rainbow*. I understand that Frank Hogg buried him in OS-9 software that is ready to be set up to run on CoCo. Speaking of Frank, we got a positive progress report on *O-PAK* from him just as we went to press. They've done their homework and should be shipping the 51 x 24 character high resolution screen by the time you read this. It looks now like you'll even be able to escape from the text mode and use the graphics commands already in CoCo's memory. This means you'll even be able to mix graphics and text on *O-PAK*'s screen. The copy routines are coming along fine also and will probably have a command something like:

OS9:XCOPY FLEX %/d0/flexname 0S9 %/d1/OS9path

Of course you'll be able to read and write Radio Shack Extended Disk BASIC disks from OS9 as well as FLEX disks.

Those Special Keys, Revisited

We picked up some new information in Des Moines about those special keys we told you about last month. "Control Q," for Quit has been changed. It is now "Control E." Now instead of "Quitting" a process, you can "End" it. This change was necessary because OS-9 Level II systems recognize the standard X-On and X-Off characters used by many terminals to tell the computer to stop sending. X-On is normally "Control Q" on these terminals so there would be a natural conflict.

And we forgot to tell you something really special about the "Control C" key. You can generate it by holding down the "Shift" key while you strike the "Break" key or, you can hold down the "Clear" key while you strike the "C." Remember when you are running OS-9 "Clear" means "Control." When you type this command you will immediately see the OS-9 prompt on CoCo's screen. But, that's only half the magic. Try this:

OS9:list filename >/p

As soon as you see that your file is being listed to your printer, type the "Control C" command. Watch what happens.

You'll notice the "OS9" prompt reappear on CoCo's screen. But isn't there something strange going on? Why is your printer still printing? You have just told OS9 to run the printing job as a background task. To prove it type the list command again but leave off the ">/p." CoCo's screen should start to fill with the listing while the printer continues to print.

Last month we said that an operating system is nothing more than a piece of software that lets you communicate —talk if you please — to many different types of hardware. It lets your printer or disk file understand what you are saying on the keyboard. It gives you a way to hook CoCo to another computer through a telephone line and modem.

Meet the OS-9 SHELL. When you talk to OS-9 by typing a command on CoCo's keyboard, you are talking to the SHELL. The SHELL is a command interpreter that translates the words you type into an action by the computer. You'll know when you're talking to the SHELL because you'll see this.

OS9:

When you see this prompt, you'll know that SHELL is waiting for you to enter a command. To do this, you simply type a command line followed by a carriage return. You can use lowercase letters, uppercase letters or a combination —the SHELL doesn't care.

Let's take a closer look at an OS-9 command line. The first thing following the prompt should be the name of a program. It can be the name of a program located in a module in your computer's memory or the name of a file that stores your program on a floppy disk.

It can be 6809 machine code that executes directly, a module containing compiled intermediate code from a higher level language like BASIC09, Pascal and Cobol or, a procedure file. Here's what happens when you give the SHELL a program name.

If it finds a module in memory with the name you have typed, it will run the program. If it doesn't find the program in memory, it looks for a disk file with that name in the current execution directory. If it finds the file, it loads it into memory and runs it.

If the name you typed is not the name of a module in memory or a file stored in the correct execution directory you still have another chance — it may be a procedure file. The SHELL knows this, and searches your working data directory for a file with the same name.

"The SHELL is a command interpreter that translates the words you type into an action by the computer."

If the SHELL finds a file in the data directory, it assumes it is a procedure file and runs it. A procedure file is a special case. Instead of holding object code that runs on your computer, or I-code that is executed by a high level language, it contains a text file that looks just like one or more command lines you could have typed from CoCo's keyboard. When the SHELL executes a procedure file, it reads text one line at a time — as if it were reading data from the keyboard. It then executes the commands in that line.

The program named the SHELL reads from your keyboard or a procedure file is usually followed by one or more parameters. A parameter gives directions to the program. It is separated from the program name by a space or spaces. For example, if you want to list a file called "Rainbow" to your terminal you must type:

OS9:list Rainbow<RETURN>

If you want a "hardcopy" of the same file you can type:

OS9:list Rainbow >/p <RETURN>

In fact, you can even send the listing to another file:

OS9:list Rainbow >CloneOfRainbow <RETURN>

Just like we've been saying, OS-9 is a very versatile operating system.

Sometimes the parameters in your command line will be options or modifiers. For example, when you want to list the names of the files in your working data directory to CoCo's screen, you type:

OS9:dir <RETURN>

Here's a way to get more information about the files. Try:

OS9:dir e <RETURN>

This command lists all available statistics about each file in the working data directory. The "e" is an option that means list the "entire" directory record. Speaking of directories and options. If you would rather see the names of the files stored in your current execution directory, type this:

OS9:dir x <RETURN>

Or, if you want to see all available information about the files stored in your current execution directory, type:

OS9:dir x e <RETURN>

Pathlists And Device Names

When you used the *LIST* command above, you were using a filename as a parameter. In this case the parameter was an abbreviated pathlist. Since you didn't pass any information about a device or directory, *LIST* assumed that the file was located in your current data directory. But, what do you do when you want to access a file that is not stored in your current data directory?

No problem, it is easy to enter a complete pathlist. A pathlist is just a description of the complete route your data must take before it arrives at its destination. It may hold the name of a mass storage file, a directory file or an Input/Output device. Microware chose the term "pathlist" instead of "filename" because you must often give the SHELL a list that contains more than one name.

For example, many "pathlists" contain a device name, and one or more directory names as well as the name of a data file. Each name in the pathlist is separated by a slash "/."

Follow these rules when you work with pathlists. Pathlists should contain names that describe three things.

1. Names of physical I/O devices
2. Names of directories
3. Names of regular files

Each of these names may be made up of as many as 29 characters or as few as one character. They must begin with an uppercase or a lowercase letter. After that they may be made up of any combination of the following characters.

1. All uppercase letter: (A—Z)
2. All lowercase letters (a—z)
3. The ten decimal digits: (0—9)
4. The underscore: (_)
5. The period: (.)

Here are some legal names:

KISSable_OS9
Rainbow
Chapter.one_and.two
ABC123

If you're wondering how OS-9 can tell the difference between a filename and a device name, here's the secret.

A device name always starts with a slash. If the device can hold multiple files — a disk drive for example — another slash followed by a directory or a filename usually follows the device name. If however the device cannot handle multiple files — a terminal or printer for example — nothing follows the device name.

Here are a few standard OS-9 device names:

Name	Device
TERM	Primary system terminal
T1,T2	Additional terminals
P	Parallel printer
P1	Serial printer
D0	Disk drive zero
D1	Disk drive one

You probably noticed these names in the list of modules we gave you last month. That's because the names above are the names of modules that hold "device descriptors." The device descriptor "TERM" for example, tells SCF, the sequential file manager, that CoCo's screen is 16 lines deep, etc. It also points SCF to the driver module CCIO so that it knows where to go to get characters from CoCo's keyboard.

Remember, if you want to name these devices in a pathlist you must type a slash before their name. Here are some common pathlists.

Group 1:	/TERM	/T1
	/p	/p1
Group 2:	/D0	/d0/cmds
	/d1/RAINBOW/First.Month	

The pathlists in Group 1 refer to devices that cannot handle multiple files.

The pathlist "/d0" refers to disk drive number zero. If you needed to know the names of the files stored on this drive, you would use this command line:

OS9:dir /d0 <RETURN>

After you hit the <RETURN> key, the names of all directories and files that you have previously saved on the disk installed in drive "d0" will be listed on CoCo's screen. Let's try another command line:

OS9:dir /d0/CMDS<RETURN>

This command lists the names of all files stored in a directory named "CMDS" located on the disk installed in drive "/d0."

Let's take it one more step:

OS9:list /d1/RAINBOW/The.Article <RETURN>

This command prints a listing of a file named "The.Article." The file is located in a directory named "RAINBOW" on the disk you have installed in device "/d1."

Advanced OS-9 Features

OS-9 has many advanced features and we'll introduce you to a few this month.

1. I/O redirection
2. Memory Allocation
3. Multitasking

During normal operation all input for your programs will come from OS-9's standard input path. Likewise all output either goes to the standard output path or the standard error output path.

Reports, listings and other data generated by your programs are usually sent to the standard output path. Error messages and various prompts are routinely sent to the standard output path. Both output paths normally send characters CoCo's screen.

When you redirect the input you tell OS-9 to get its input somewhere else. Likewise, when you redirect the output you tell OS-9 to send its data somewhere other than the screen. For example, when you sent the directory listing to the printer earlier in this chapter, you were redirecting the output to the printer.

There are three redirection operators that you will use when you give commands to the SHELL:

<	means	redirect the standard input path
>	means	redirect the standart output path
>>	means	redirect the standard error output path

There are many ways to use these operators. You may redirect the input to your program from another terminal plugged into CoCo's RS-232 jack or from a modem. Or, you may send output to a disk file for later printing. There is no end to the possibilites.

About Memory

Some OS-9 programs need very little memory to run. Others require thousands of bytes. This is not a problem however because the header of each program module tells OS-9 the minimum amount of memory needed to run a program. However, when you need more memory, it is an easy matter to request more with OS-9's memory size modifier. There are two ways to do this.

```
OS9:copy #8 myfile yourfile
OS9:copy #2K hisfile herfile
```

The first command above tells OS-9's copy utility to use eight 256-byte pages of memory — a total of 2048 bytes. And believe it or not, the second example also gives the copy command 2048 bytes to use. It is requesting two "k" or two thousand bytes of memory.

There are also several ways to run a series of OS-9 programs. You can run them sequentially — one after the other; you can run them concurrently — all at the same time; or, you can synchronize them so that the output of one feeds the input of another using OS-9's pipes.

There are two ways to run programs sequentially. You may type one command line followed by a carriage return, wait for the program to finish and then type the next command line — or, you may type more than one command on a line. You must use a semi-colon to separate the commands if you chose the second method. Here's an example:

```
OS9: copy hisfile herfile ; dir >/p <RETURN>
```

This command will copy the file named "hisfile" from the current data directory to a file named "herfile" in the same directory. It will then immediately print a listing of the current data directory on your printer.

If you want to run more than one program at the same time you must ask OS-9 to execute the programs concurrently by using an ampersand, "&."

You may run any number of programs at the same time. The CoCo's memory will be the only limiting factor.

Pretend for a moment that you have just finished an assignment in school. You need to print it so you may turn it in to your instructor, but at the same time you need to be working on another term paper. To do both jobs at the same time, try this!

```
OS9:list EnglishII.Assignment >/p&&004
OS9:edit Term.Paper ____ History
```

Just like magic, the printer will start right after you hit the [ENTER] key. Yet, the familiar OS-9 prompt will pop on CoCo's screen almost immediately. As soon as it appears you can type the next command line and start the editor. The printer will run as long as it needs to print the English assignment. It won't bother your editing at all.

Logging On a Timesharing Terminal

With OS-9 you can do more than just print one file while you are editing another. One of the major uses for concurrent execution is terminal timesharing. For example, you could use your editor to write a news release about a new product using CoCo's keyboard and screen while someone else in the family runs a BASIC09 program to balance the check book. First, plug a terminal into CoCo's RS-232 jack, then type the following:

```
ON COCO:

OS9:tsmon /t1&&005

OS9:

YOU'LL SEE THIS ON THE OTHER TERMINAL

OS-9 Level I Version 1.0 Timesharing System 8/19/83 21:30:35

Use name? esther

Password:

Process #5 logged 8/19/83 21:31:36

Shell

OS9:
```

CoCo has prompted you to go back to work. But, while you're writing that news release, your wife can probably finish the check book. CoCo will pay for itself in no time.

When you first run the timesharing monitor program, TSMON, nothing happens. The terminal remains idle until someone hits its return key.

Also, when using a terminal with CoCo, you must log on. To do this you enter your name and the proper password. You will need to give everyone in the family passwords before they try to log on the first time. If they don't know the magic word, OS-9 won't allow them to compute.

To log off a CoCo timesharing terminal, you need only hit the ESCAPE key. This represents an end-of-file signal and returns your terminal to an idle state.

You Can Feel Secure

While timesharing CoCo with OS-9 you won't need to worry about someone else writing in your data files. The system protects you with its file security system.

Each OS-9 directory and file has several attributes that tell the system who owns the file and who may use it. They are:

1. Write permission for owner.
2. Read permission for owner.

3. Execute permission for owner.
4. Write permission for public.
5. Read permission for public.
6. Execute permission for public.
7. A "sharable" attribute.
8. A directory attribute.

Let's explain the special cases first. If the "sharable" attribute is turned on, OS-9 will not let two users use a file at the same time.

The directory attribute tells OS-9 that a file is a directory file. A directory file is special because it cannot be changed by the user. To change a directory or delete it during an operation would create total havoc with the file system. In fact, there would no longer be a system.

The other file security attributs almost explain themselves. They work because OS-9's file system automatically stores the user number associated with a process when it writes a file. If you are the owner of a process, you will own any files it creates.

If you CREATE a file with none of the public attributes set, you will be the only person that can READ, WRITE or execute that file. You may even ask the system to protect a file from you. For example, after getting a mailing list in final form, you may clear both the public and owner WRITE permission attributes to prevent accidental deletion or modification.

Summary

By now you should be fired up and ready to go. Run through the SHELL command lines we used this month and practice with your own. Don't be afraid to make a mistake. The worst thing that can happen is an ERROR #XXX message.

Next month, thanks to Rich Hogg, we'll show you how to change a byte in memory here and there to speed up your disk drives and if you are using double sided drives, we'll show you how to make the opposite side of drive/d0,/d2, etc. We'll also introduce you to pipes and show you how to copy and entire directory with one swift command, via a "pipe." We would pass along the information now but we're not sure the advance version of OS-9 that Tandy provided *the Rainbow* is the same as the Production version. As soon as we can test our changes on a production version, we'll print this information.

Ham And Cheese (On Rye?)

Our technical "big cheese," Dan Downard, who is a pretty hot ham radio operator, as well, provides us with this list of radio nets devoted to the Color Computer.

Time	Frequency	Days	Net Control
1930Z	7.228MHz	Sunday	AF4K-Brian
2000Z	14.275MHz	Sunday	WB3EBA-Al
2100Z	7.260MHz	Sunday	WB3EBA-Al
0300Z	7.230MHz	Monday*	W8UNB-Mel

*-Sunday Night

— Dan Downard

The Results Are In...

In deciding the winners of the Rainbow's Simulation Contest, one judge finds it all to be a very real and dazzling experience.

By Charles Springer

Someday soon a bleary-eyed individual may sit down and write a simulation program about what it was like to be chief judge of *the Rainbow's* first annual Simulation Contest. It should be a good one because it will include a dazzling array of graphics, super sound effects and dozens of mind-bending options.

Other special effects will include getting buried under stacks of cassette tapes and disks and dozens of pages of instructions. He'll be forced to purchase a new color TV set when his old one suddenly loses the ability to distinguish green from red and/or blue. There'll be visits to a friend's house to have a disk program converted to cassette. There will, of course, be more I/O errors and seemingly dozens of *PCLEAR1* statements. And, if he's as fortunate as I am, has a very understanding wife! Plus, friends who, when asked for their opinion, will not keep changing their minds.

But the real joy will be in sampling a wide variety of simulation programs, the quality of which leaves him in awe, anticipating what kind of programming developments must surely lie ahead for CoCo lovers in the very near future. Via his dream machine, he will: battle the Civil War and see the world destroyed during World War III, travel to the Moon, to Mars and beyond, conducting experiments along the way; go bankrupt running a restaurant, make a million as a manufacturer, and break even as a clothing store proprietor; run for President of the United States; take responsibility for mid-air collisions as an air-traffic controller; drown while learning to sail or taking part in a colonial sea battle; and experience the thrill of victory in the seventh game of baseball's World Series.

He will, at times, experience exhaustion. He will be persistent, however, playing the programs over and over, making notes of the fine points, the strong points and the weaknesses. And in the end, like they say, the cream will rise to the top, the judges will reach consensus, and the winners will be rewarded.

Entries were received from readers across the country, from such places as Solana Beach, California; Jupiter, Florida; Crivitz, Wisconsin; Downers Grove, Illinois; and Mt. Pleasant, Pennsylvania. With seven entries, Illinois captured state honors for the most submissions. There were five each from New York and Iowa, four from Pennsylvania, and three each from California and Wisconsin.

The Rainbow's desire for fairness mandated that we establish beforehand a uniform set of guidelines by which each category would be judged. They included: 1) clarity of instructions, 2) ease of loading, 3) format, 4) vocabulary, 5) grammar, 6) creativity, 7) enjoyment, 8) realism, 9) resolution, 10) use of color, 11) responsiveness, 12) use of sound, 13) level of challenge, and surprises.

Tough? They are meant to be because if you ever attempt to market one of those gems you've created on your kitchen table, you'll find that the general buying public's demands will be even harder to satisfy. Plus, if CoCo programs are going to be used to help educate our youngsters, we as creators must pay careful attention to such things as vocabulary, grammar and spelling — not listed among the criteria but most definitely a consideration. We also took note of historical accuracy where appropriate, the overall educational value of an entry, and the degree of interaction that a program allowed between the computer and its user.

Then, of course, there are the built-in prejudices that a judge has no matter how objective he or she attempts to be in such a situation. And there are those oh-so-subtle little things that can make a difference in the quality of a program, particularly in a simulation program.

Okay, enough sermonizing. There's no getting around the fact that not everybody can win in a competitive situation. So, here goes . . .

Through a grueling process, which involved many reloadings and endless comparisons, we determined that the following 15 entries, which are listed in alphabetical order, are the very best in a very healthy field. We have singled out the best of show, the runners-up and a number of honorable mentions:

BRADD BALASCO of Mansfield, Massachusetts, for *Sailing.*
DON BRADFORD of Solano Beach, California, for *Civil War.*
RICHARD BROWN of Lorain, Ohio, for *Ship of the Line.*
GREGORY CLARK of Syracuse, New York, for *Flood.*
MARTIN LEE of Overland Park, Kansas, for *Baseball.*
DAVID MANN of North Syracuse, New York, for *Restaurant.*
GEORGE MAREK of Glastenburg, Connecticut, for *Air Traffic Controller.*
F. W. McKENZIE of Saugerties, New York, for *Flight.*
GILBERT S. MURRAY of Columbus, Ohio, for *Big Business.*
PAT PUGLIANO of Baden, Pennsylvania, for *Street Racer.*
DAVID ROSICKY of Pittsburgh, Pennsylvania, for *Street Racer.*
GEOFF STARK of Winnepeg, Manitoba, Canada, for *Mars Sim.*
JEFF STEVENS of Columbus, Ohio, for *Congress.*
ROBERT K. TYSON of Jupiter, Florida, for *Election '84.*
TOM WEBER of Milwaukee, Wisconsin, for *War Games.*

For their efforts all of these folks will receive prizes from *the Rainbow's* generous advertisers, and each of them can expect to see their entries published in book form in the near future.

In keeping with a *Rainbow* tradition that started with the Adventure Contest winners announcement in January — and a format that is followed in the Miss America competition — the grand finale will come later. But first some special awards and honorable mentions, then the runners-up, followed by the top two winners whose programs are listed in this month's issue.

The Up, Up And Away Award goes to F.W. (Skip) McKenzie for *Flight,* a delightful program that provides a panorama of mountains, trees and city skylines for you to negotiate, using the arrow keys on your keyboard. Skip's entry was fun to play and there was minimum use of confusing indicators on what are usually difficult-to-read instrument panels. The view from the cockpit separated this one from the average flight simulation.

The Water Over The Bridge Award to Gregory Clark for *Flood.* Due to heavy rain, Rainbow River overflows its banks. Greg provided us with a very detailed and colorful map in high resolution to track the river's progress, along with timely weather forecasts and gauge readings of the water level. The challenge was to use the floodgates, the heavy equipment (trucks and bulldozers), the sandbags, fuel supplies and manpower available to us to quell the ever-constant threat.

The John Paul Jones Award goes to Richard Brown, a freshman at Dartmouth College, for his *Ship O' Line,* featuring sea battles as the one between the U. S. Frigate Constitution and the British ship Guerrier, which almost ended the War of 1812. A 17-page set of instructions, complete with bibliography, will prepare you for the stiffest foe on the seas. Separate screens allow you to keep track of your maneuvers, and there are dozens of options available that have obviously been carefully researched, lending a great deal of authenticity to this sterling effort.

The Union Label Award to Gilbert S. Murray for *Big Business,* in which you are the manufacturer of Enertabs, a

new system of energy storage. Your contract is for between six and 10 years, which are divided into quarters. You're in competition with nine other companies, which can be represented by CoCo or, believe it or not, nine other players of your choosing. The program establishes a business cycle, includes predictions about the economy and determines the demand for your product. A good program for teaching economics to would-be entrepreneurs.

The Sound Of Music Award goes to David Rosicky, a young high school student, for the funeral music following the demise of "Al E. Katt" who is wasted on his way home with his paycheck. David also receives the **Best Use Of Special Effects** for his title page, which features the name of the program *(Street Races)* racing across the screen to puttering sound, and intermittent use of checkerboard pattern that can only be described as hypnotic.

The Chapter Eleven Award goes to David Mann for *Restaurant,* a simulation that has you inheriting the family business. Your job is to run it and turn it into one of the largest chains in the country. Hiring, firing, buying, selling and advertising are all your responsibility. The problem I had was that I didn't know the first thing about buying food by the pound and setting an appropriate price — so the first 10 times through I went bankrupt. You also have to pay your employees a decent wage or they'll leave. You have a choice of three levels of competition — corporate, partnership, or entrepreneur. The opening graphic of a table set for two was nicely done, creating the appropriate atmosphere.

The Golden Glove Award goes to Martin Lee for *Baseball,* a simulation of the seventh game of last year's World Series between the Milwaukee Brewers and the St. Louis Cardinals. A good baseball game for CoCo is long overdue (so is a good basketball game for that matter) and this one is a good start. Featuring the starting lineups in that game and basing their hitting and pitching probabilities on their actual averages, the game is uncannily consistent with what you might expect in actual competition. Between each batter you are given offensive and defensive option — and you can bet that the Cards' Willie McGee will usually be successful in stealing second base. I was a little suspicious when Milwaukee won the first five games, but then St. Louis got hot and won the next three. There are high-scoring affairs and there are pitchers' duels. The old favorite, "Take Me Out To The Ballgame," is a nice touch between innings, as is the scoreboard. If you like baseball, you'll enjoy this one.

The If You Walk, You're Fired Award goes to George K. Marek for *Air Traffic Control,* in which you are responsible for the safety of hundreds of air passengers. The display provides information from radar and the ground computers about aircraft location, speed and status. This one requires a lot of concentration, as well as attention to the instructions, because you can't leave anything to chance in the role you are playing. Mid-air collisions and crash landings are common occurrences at first, but as you become accustomed to the pace the odds improve significantly. This is a well-conceived simulation, nicely packaged and with a lot of potential.

Fifth Runner-Up is Bradd Belasco for *Sailing,* a must for those of us who have the urge to head for the lake over the weekend. Dividing the screen into three sections — the top left for a side view of sail status, the top right an overhead view of the board to show direction of the board, the boom and the rudder, and bottom half of the screen to display the navigational course. Using your joystick to determine the proper setting, you must make it to the other side of the lake and dock safely. If you make a mistake, the graphic changes to a sinking sailboat and a man treading water. You are given a danger signal if you place the boom in such a position that the wind will cause to quickly swing around and send you sailing like an opening kickoff, but you're also given time to respond. Bradd kept this simple and easy to understand, with exceptional graphics. A fun learning experience.

Fourth Runner-Up is Pat Pugliano for *Life With CoCo,* a simulation with which many *Rainbow* readers can readily identify. It's about a computer programmer who is trying to make it big by creating games and writing articles for computer magazines. But first he has to learn assembly language and build up his inventory of equipment. Like many of us he reads the ads to keep up with sales on software. He visits a computer store and, if he has used his money wisely, he purchases a utility program. The real problem for our hero is that there are only so many hours in the day for him to do his regular job, read computer magazines, work on his programs and articles. Sometimes he neglects his diet and his sleep because he is so attached to CoCo. If he manages his time, health and money wisely, he is a tremendous success and all of us are buying his programs.

Third Runner-Up is Don Bradford for *Civil War,* a situation in which you are commander of the entire Southern Army, reporting directly to Jefferson Davis, President of the Confederacy. This game is loaded with options, multiple choices and a nice map of the North and South pinpointing the battle in which you are currently engaged. You receive your instructions from the President via telegram which is transmitted staccato style across the screen, telling you where to do battle next. Before the battle, however, you are required to submit a formal budget to the President. Unless you go in the red, he will usually approve it. You are given six options for the battle plan (including the ability to surrender the battle or to give up the war). Just before entering the skirmish, you will usually receive a report from your spies, unless they have been captured. After the battle, you are given a report of casualties, desertions, prisoners-of-war and whether you won or lost. You also can see a record of all battles. After 13 battles, a winner is declared (usually the North since CoCo is its commander). If you lose, you must unconditionally surrender all of your territories, admit your guilt and wrong-doing, be tried as a war criminal, and hanged! Great fun and a good review of this historic period.

Second Runner-Up is Geoff Stark for *Mars Sim.* Geoff, a 15-year-old student, has done a tremendous job of creating an action-packed simulation that allows you to do a visual scan of the planet's surface, unless you land at night, at the beginning. There are different directions and entirely different views. Next, you are able to maneuver a robotic arm that enables you to scoop up samples and return them to the lab for examination. In the lab, expose the sample to oxygen, Earth light, heat, and water to determine whether life exists. A diagnostic function lets you check all of the systems aboard the ship. Another option allows you to drill up to 1.5 meters testing the planet's interior. Geoff's documentation is extensive and indicates just how rapidly some students have advanced in programming. Take a bow, Geoff, and stay with CoCo because we're expecting even better things from you.

First Runner-Up is Jeff Stevens for *Congress.* A high school social studies teacher, Jeff uses this program in a government course to show how a bill passes through Congress and becomes law. It opens with a Presidential Application, complete with dotted lines, which you must complete to assume the office. Your object is to go down in history as a

great President (sound familiar?) and your ability to get your programs passed through Congress with a minimum number of amendments and special favors will determine how effective you will be. You are given two options for major programs, which are strikingly similar to the approaches taken by recent Democratic and Republican candidates. The song, "Hail to the Chief," opens the game and you're treated to other patriotic tunes when a bill passes both houses. The visual effects are enhanced by digital counters which keep track of "yes" and "no" votes. When the bill reaches your desk, Mr. President, you have the option of veto (because of undesirable amendments), signing it into law, or letting it take effect without your signature. Then, you go on to another major priority of your administration, following the same steps. You have only a limited time, of course, to get your program through Congress and your eventual stature depends upon your ability to get all your programs passed during one term. A warning to Republicans: both the House and Senate are usually heavily loaded with Democrats, so you will experience a lot of problems. A Gallup Poll is taken after your term and, if you do poorly, you will probably be as popular as Herbert Hoover after the stock market crash. On the other hand, you may be compared to Franklin D. Roosevelt. This one was among the most fun to play, while being very educational. Jeff Stevens' government classes must be in great demand!

And now for the Best of Show, the Grand Prize winners in the Graphic and Non-Graphic categories!

Graphics Best Of Show goes to Tom Weber, a freshman at Marquette University, for *War Game*, a totally involving simulation, which makes the movie by a similar name seen rather tame. Tom plans to major in computer science at Marquette and, judging from his entry, will put the school on the map for something other than its basketball team (sorry Marquette, but being a Louisville fan, I couldn't resist the dig).

Tom says he is self-taught in many aspects of computer programming. He is quite active in CoCo-MUG — the Color Computer Milwaukee Users' Group — in which he serves as a member of the Executive Committee.

He obviously knows what he is doing when he sits down in front of his CoCo because *War Game* is a highly sophisticated undertaking, which has been assembled in a relatively short period of time since the smash-hit movie started making its rounds.

While the subject of his simulation is rather gruesome, it's no secret that computer war games hold immense fascination for the majority of us whether we want to admit it or not. Tom's program, which loads in four parts, is, as far as the judges can tell, a realistic version of what can happen when and if some misguided power monger gives the command to start pushing buttons. It ranks high as an educational tool in that respect, as well as in loading complicated programs — so please pay close attention to the loading instructions after you have typed in the program. He says that it will not run on disk because of memory requirements — but that's never discouraged some CoCo users we all know. Tom, you're a winner. Good luck at Marquette, and keep those contributions coming to *the Rainbow*.

Non-Graphics Best of Show And Grand Prize goes to Dr. Bob Tyson for *Election '84*, a simulation depicting the last 100 days of next year's presidential campaign. If the name Bob Tyson rings a bell, it's because he's the same person who created *Strategy Football*, which appeared in *the Rainbow*'s August issue. Bob is a Senior Systems Engineer at United Technologies Research Center in West Palm Beach, Florida, where he works on high energy laser beam control system development for use in defense and communication systems by our country's armed forces.

He started doing the winning program featured in this issue "because I wanted to do something that was relevant and timely, and because I have always had a fascination for politics." Bob also says he developed a hurricane tracking system a few years back for his CoCo, a program which undoubtedly would have great utility in Florida.

He's a graduate of Penn State University, where he probably picked up some of the finer points of the game of football that were used in his August simulation. He also learned Fortran at Penn State, a skill that he says served him well in the Navy, and in development of the obviously strong skills demonstrated in his latest effort. Bob says his wife Sue was a great help in *Election '84*, although she was never able to sit down in front of the computer. "She has her hands full taking care of our five-year-old daughter Kia and two-year-old son Andy," he notes.

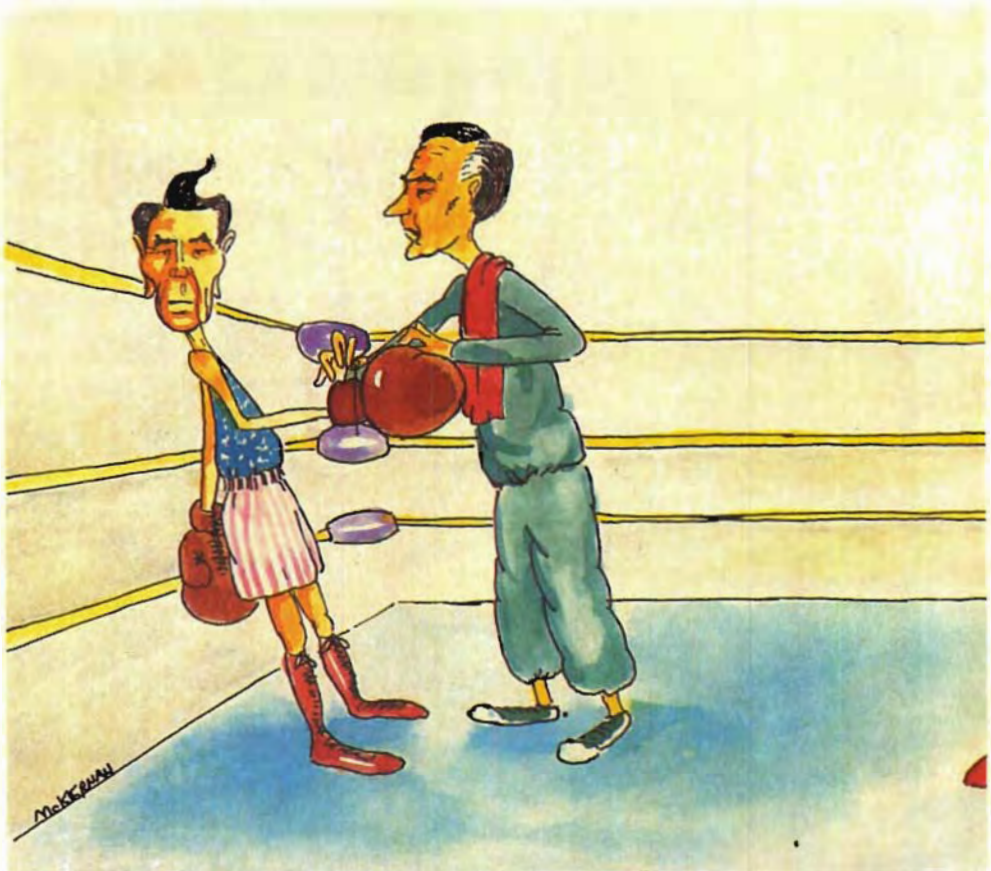
Bob says he has been playing around with the idea of doing a simulation on the 1984 Olympics. We hope he does such a program because Bob obviously knows what he's doing and we'd like to be able to share another of his programs with our readers.

You will enjoy *Election '84* because it includes all of the aspects that make Presidential campaigns so fascinating. And the suspense as the popular and electoral votes are being tabulated will make all of the time that you have spent typing in this great simulation well worth every minute. We think you'll agree that *Election '84* is among the best in the field.

And the winner is...

Election '84 is a simulation of the 1984 presidential election campaign. The simulation is performed (played) by one or two persons representing the Democratic and Republican parties. In the one-player simulation, the computer will handle the affairs of the opposition party.

Election '84 allows each player (the campaign manager) to handle his or her candidate's affairs during the campaign. The campaign begins with 100 days left before the election. A candidate can travel to different states and campaign; he can advertise in any of the states; he can challenge his opponent to a debate; he can raise funds for further campaigning; or he can yield his position (and just rest where he is). At any period where the above options are presented, a campaign manager can check his "intelligence" and find the latest Gallup Poll readings, the latest news reports, and information about travel history of all of the candidates. Each campaign manager controls both his presidential and vice-presidential candidates separately. At the end of the campaign, on election day, the campaign managers just sit back and watch the election returns roll in from all across the country. A tally of popular and electoral votes is kept, and before long, there is a new man in the White House. Then you can figure out where you went wrong, or pack your bags for the staff job you earned. Maybe President Reagan is reelected; it's up to you.



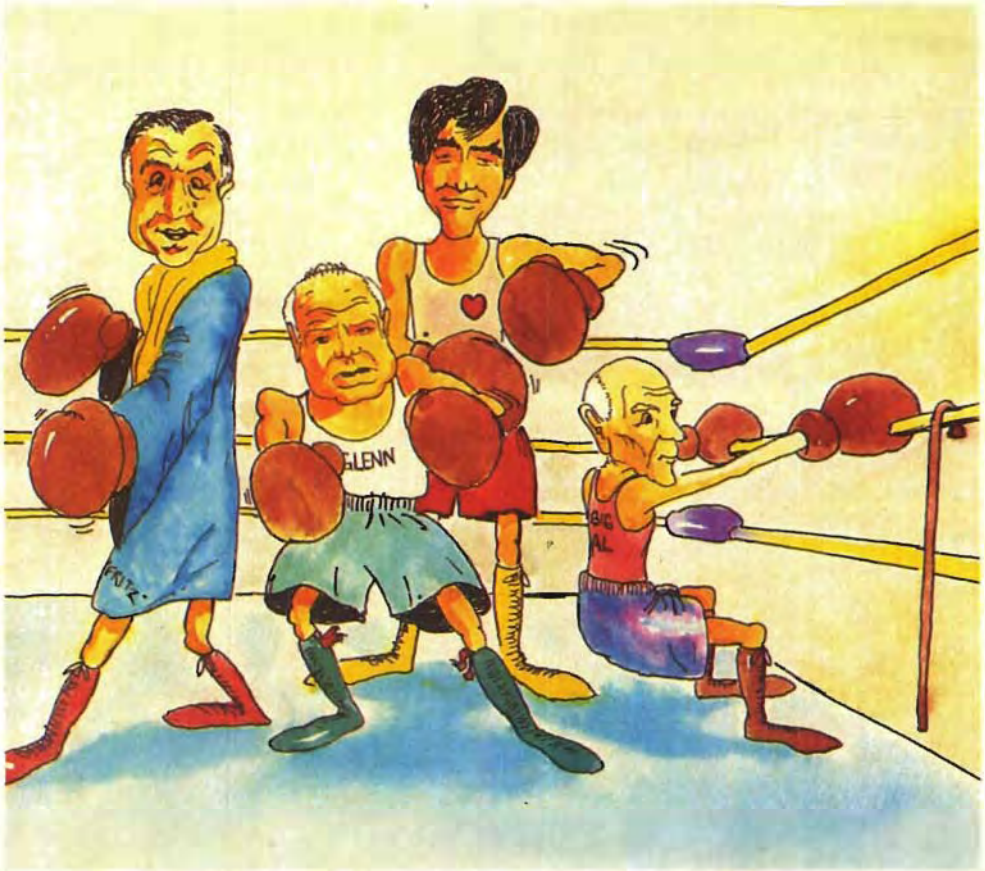
Simulation: The campaign begins with your choice of a long or short campaign and then the selection of the candidates. Choose the short campaign for the first time through, the longer one for more realism. Even if only one player is involved, all four candidates (two presidential and two vice-presidential) must be chosen. The program gives six possible candidates for the positions on the ticket from each party. These are the most likely candidates from events this summer (1983). Points for financial strength, experience, and charisma are assigned. Changes can be made if events change between now and the summer of '84. (See below for program description). After candidate selection, the campaign begins with alternating turns from the four candidates. If the computer is handling the affairs of one party, decisions will be made very quickly, but some require your reply, so watch carefully.

If you have to make the decisions, just follow the menu. This is where the fun comes in. If you want to find out where everybody has been, just hit "I" for intelligence. You will see a screen display with the two letter identification for each of the 50 states and Washington, D.C. (yes, Washington D.C. has three electoral votes). Beside the state i.d. is

(Robert Tyson holds a doctorate in physics and designs and analyzes high energy laser beam control systems using computer simulations. He and his family live in Jupiter, Florida.)

Election '84

By Bob Tyson



its electoral votes. Also alongside, you may see either blue or red graphics characters. If the upper red is colored in, then the Democratic presidential candidate has been there; if the color appears in the lower red block, then the Democratic vice-presidential candidate has been there. If the upper block is blue, the Republican presidential candidate has visited, and if the lower block is blue the visit was by the Republican Vice-Presidential candidate. Get it? Democratic party is red; Republicans are blue. Upper is Presidential; lower is Vice-Presidential.

If you continue, you will get the current Gallup Poll for the election. Remember, the Poll only tells popular vote count; it takes electoral votes to win the election, so use the "intelligence" feature whenever necessary (or just to find out a state's identification). Oh yes, the latest news report may be of interest. Read it carefully and act accordingly.

You can also campaign in any state. Just press 'T' for travel and follow the prompts. Travel costs money, so be thrifty unless you're on an all-out blitz near the finish. You do not need to travel to pick up votes. You can press 'A' and advertise in any state. This costs a lot of money, but it's well worth it. Many votes can be had for a few TV commercials. When your total campaign treasury total is below zero, you will be forced to raise money. This may be at an inopportune time, so you may want to raise money early in the campaign. If your total goes negative near the end of the campaign, that's okay, since everybody goes into debt running for the Presidency. You and your creditors will hope that you can pay it back.

If you press 'Y' and yield, you will pick up more votes wherever you are. It's a good rest. A high risk, but high payoff, option is the debate, 'D.' You can pick up thousands or millions of votes across the country by just this one event. But be careful, a debate must be accepted by your opponent and one slip of the tongue could ruin your chances.

When election day arrives, just sit back and watch. You may be sorely disappointed as your candidate loses tough states by a few votes. But then again you may be overjoyed when he wins by a landslide.

How the program works: Setup takes place in lines 5—102 with calls to subroutines to run the logo, the candidate selections, etc. A large iterative loop extends from 200—340 where the bulk of the simulation is run. Subroutine 7800 creates the displays for the option selections. The arrays dimensioned in lines 10 and 11 hold the state identifiers (S$), the votes cast, VT, the visits, VS, the candidate strengths, SR, and the money, MY. The other variables are flags and codes to keep track of where the simulation is and where it hasn't been.

Changes can be made to update the simulation. 4500—4601 contain the news stories. 8010—8011 contain the Democratic candidates and 8050—8052 contain the Republicans. If you change the

names, strengths, or homestates be sure to keep the characters and strengths aligned. The SR matrix 'reads' the strengths from the character strings. Lines 9010 and 9011 are the time delays. If they are too short or too long, just change the length of the FOR/NEXT loop. Lines 8959—9003 contain statistical data on historical voting patterns, advertising costs, and total popular votes cast. Don't alter these if you don't want to change the history of the world!

Well, that's it. If you want to be a power politician, this simulation will let you. "Happy days are here again. . . "

60.....0340	5130... 22EF
310.....0554	6521....2717
584.... 0B28	7500...2BAD
822.... 0FC2	7809... 2EF7
2000....1413	8017... 34E0
4561... 1D24	8095....3956
	END...3DC3

The listing:

```
5 GOTO10000
10 DIMS$(51),VT(51,2),VS(51),SR(
2,2,3),MY(2)
11 DIMC$(6),CS$(6),PC$(2,2),HS$(
2,2),CC(2,2),FG(15)
12 H$="###,###,###"
15 FORI=0TO50:VS(I)=0:FORJ=1TO2:
VT(I,J)=0:NEXTJ,I
20 FV=0:FORI=1TO15:FG(I)=0:NEXT:
GOSUB2000
25 RESTORE:FORI=1TO255:READX:NEX
T:FORI=0TO50:READS$(I):NEXT
28 CLS:PRINT@128,"ARE YOU READY
FOR A SHORT OR A  REGULAR CAMPAI
GN?                ...   TYPE AN
 <S> OR AN <R>"
29 K$=INKEY$:IFK$="S"THENST=10EL
SEIFK$="R"THENST=3ELSE29
30 CLS:PRINT@132,"SELECT MODE:
                  1 CAMPAIGN MAN
AGER OR           2 CAMPAIGN MAN
AGERS?"
35 INPUTMD:IFMD=1ORMD=2THEN36ELS
E30
36 ONMDGOTO37,40
37 CLS:PRINT"CHOOSE YOUR POLITIC
AL PARTY     'R'EPUBLICAN OR 'D'
EMOCRATIC"
38 K$=INKEY$:IFK$=""THEN38ELSEIF
K$="R"THENMD=11ELSEIFK$="D"THENM
D=12ELSE38
39 GOTO50
40 PRINT@256,"PLAYER 1 IS THE DE
MOCRATIC PARTY AND PLAYER 2 IS R
EPUBLICAN":GOSUB9011
42 FORI=0TO50:VT(I,3-PL)=RND(5)+
VT(I,3-PL):NEXT:GOTO680
50 FORJ=1TO2
55 GOSUB8000
60 ONJGOSUB8010,8050
70 NEXT
100 DY=100:PL=1:CN=1
102 FORJ=1TO2:MY(J)=250000*(SR(J
,1,1)+SR(J,2,1)):NEXTJ
200 FLAG=1:IFDY=0THEN700
210 GOSUB7800
220 IFK$="T"THENGOSUB5000ELSEIFK
$="A"THENGOSUB5500ELSEIFK$="R"TH
ENGOSUB6000ELSEIFK$="D"THENGOSUB
6500ELSEIFK$="Y"THENGOSUB7000ELS
E210
230 FLAG=0:CN=2
240 GOSUB7800
250 IFK$="T"THENGOSUB5000ELSEIFK
$="A"THENGOSUB5500ELSEIFK$="R"TH
ENGOSUB6000ELSEIFK$="D"THENGOSUB
6500ELSEIFK$="Y"THENGOSUB7000ELS
E240
260 PL=2:CN=1
270 GOSUB7800
280 IFK$="T"THENGOSUB5000ELSEIFK
$="A"THENGOSUB5500ELSEIFK$="R"TH
ENGOSUB6000ELSEIFK$="D"THENGOSUB
```

```
6500ELSEIFK$="Y"THENGOSUB7000ELS
E270
290 CN=2
300 GOSUB7800
310 IFK$="T"THENGOSUB5000ELSEIFK
$="A"THENGOSUB5500ELSEIFK$="R"TH
ENGOSUB6000ELSEIFK$="D"THENGOSUB
6500ELSEIFK$="Y"THENGOSUB7000ELS
E300
320 DY=DY-ST
325 IFDY<30ANDST=10THENST=5
330 IF(DY<10ANDFV=0)THEN335ELSE3
40
335 FV=1:ST=1:FORI=0TO50:VT(I,1)
=INT(VT(I,1)/2)+RND(2):VT(I,2)=I
NT(VT(I,2)/2)+RND(2):NEXT
340 PL=1:CN=1:GOTO200
500 X=RND(7):GOSUB9010:GOSUB9010
510 IFMY(PL)<0ANDDY>3THENX=8
520 ONXGOTO530,550,550,570,570,5
70,580,590
530 CLS:PRINTPC$(PL,CN):PRINT"IS
 CHALLENGING YOU TO A DEBATE  ON
";:IFRND(2)=2THENPRINT" ECONOMIC
S."ELSEPRINT" DEFENSE."
532 PRINT"DO YOU ACCEPT? (Y/N)"
534 K$=INKEY$:IFK$="Y"THEN536ELS
EIFK$="N"THEN538ELSE534
536 X=RND(2):GOSUB9010:GOSUB9010
:IFX=PLTHENPRINT"THE WASHINGTON
POST EDITORS FEELTHAT YOU WON TH
E DEBATE."ELSEPRINT"THE PRESS CO
RPS AGREES THAT YOU LOST THE DEB
ATE."
537 IFX=PLTHEN540ELSE542
538 GOSUB9011:GOSUB9011:X=RND(2)
:IFX=PLTHENPRINT"YOU GAINED SOME
 SUPPORT FOR YOURSTAND."ELSEPRIN
T"YOU LOST THE CONFIDENCE OF YOU
R PARTY REGULARS"
539 IFX=PLTHEN540ELSE542
540 FORI=0TO50:VT(I,PL)=RND(5)+V
T(I,PL):NEXT:GOTO680
542 FORI=0TO50:VT(I,3-PL)=RND(5)
+VT(I,3-PL):NEXT:GOTO680
550 CLS:X=RND(51)-1:IFPL=1THENPR
INT"THE DEMOCRATIC PARTY"ELSEPRI
NT"THE REPUBLICAN PARTY"
551 PRINT"IS ADVERTISING IN -";S
$(X);"-"
552 GOSUB9011:RESTORE:FORI=1TO15
3:READY:NEXT
553 Y=0:FORI=0TOX:READY:NEXT
554 Y=Y*50000:MY(PL)=MY(PL)-Y:VT
(X,PL)=VT(X,PL)+RND(15)+15
555 GOTO680
570 CLS:X=RND(51)-1:PRINTPC$(PL,
CN):PRINT"IS CAMPAIGNING IN -";S
$(X);"-":GOSUB9011
572 RESTORE:FORI=1TOX:READX1:NEX
T
573 RESTORE:FORI=1TOCC(PL,CN):RE
ADX2:NEXT
574 RESTORE:FORI=1TO51+X:READY1:
NEXT
575 RESTORE:FORI=1TO51+CC(PL,CN)
:READY2:NEXT
576 X3=(X1-X2)*(X1-X2)+(Y1-Y2)*(
Y1-Y2):MY(PL)=MY(PL)-900*X3:CC(P
L,CN)=X:VT(X,PL)=VT(X,PL)+33-7*C
N+RND(8)+2*SR(PL,CN,3)
577 LO=CC(PL,CN):GOSUB1000
578 GOTO680
580 CLS:PRINTPC$(PL,CN):PRINT"IS
 RESTING IN -";S$(CC(PL,CN));"-"
:GOSUB9011
582 VT(CC(PL,CN),PL)=VT(CC(PL,CN
),PL)+20-5*CN+RND(5)
584 MY(PL)=MY(PL)-1000
586 GOTO690
590 CLS:PRINTPC$(PL,CN):PRINT"IS
 HOLDING A FUND RAISING DINNERFO
R PARTY REGULARS":GOSUB9011
592 X=RND(3):Y=SR(PL,CN,1)*10000
*RND(INT(10/X)):VT(CC(PL,CN),PL)
=VT(CC(PL,CN),PL)+20-5*CN*RND(5)
594 MY(PL)=MY(PL)+Y
680 GOSUB9010
690 IF(PL=1ANDCN=1)THEN230ELSEIF
```

```
(PL=1ANDCN=2)THEN260ELSEIF(PL=2A
NDCN=1)THEN290ELSEIF(PL=2ANDCN=2
)THEN320
691 GOTO230
700 CLS:PRINTSTRING$(41,CHR$(239
));" ELECTION DAY ";STRING$(41,C
HR$(239))
702 FL=0:MY(1)=0:MY(2)=0:X1=0:Y1
=0
709 PRINT"THE RETURNS ARE COMING
 IN FROM  THE EAST:"
710 FORI=0TO11
711 GOSUB800:NEXT
712 CLS:PRINT"THE RETURNS ARE CO
MING IN FROM  THE SOUTH"
713 FORI=12TO23
714 GOSUB800:NEXT
715 CLS:PRINT"THE RETURNS ARE CO
MING IN FROM  THE MIDWEST"
716 FORI=24TO36
717 GOSUB800:NEXT
718 CLS:PRINT"THE RETURNS ARE CO
MING IN FROM  THE WEST"
719 FORI=37TO50
721 GOSUB800:NEXT
722 IFI>=50THENFL=0
724 CLS:GOTO860
800 GOSUB9011:CLS:PRINT"CURRENT
POPULAR VOTE TOTALS:":PRINTLEFT$
(PC$(1,1),22):PRINTUSINGH$;X1:PR
INTLEFT$(PC$(2,1),22):PRINTUSING
H$;Y1:PRINTSTRING$(32,CHR$(185))
:PRINT@192,"RETURNS COMING IN ..
.":GOSUB9010:PRINT@224,"STATE ..
 ";S$(I);
805 SOUNDI*4+1,3:RESTORE:FORJ=1T
0205+I:READY:NEXTJ
810 PRINT",";Y;" ELEC. VOTES"
815 RESTORE:FORJ=1TO255:READX:NE
XT:FORJ=1TO51:READK$:NEXT
816 FORJ=0TOI:READX:NEXT
817 J=VT(I,1)+VT(I,2):IFJ=<0THEN
J=1ELSEJ=INT(J)
818 T=VT(I,1)/J:W=VT(I,2)/J
820 T=1500*X+7000*X*T+RND(15000)
:W=1500*X+7000*X*W+RND(15000)
821 X1=X1+T:Y1=Y1+W
822 IFT=WTHENW=10000-RND(20000)+
W
823 IFT=WTHEN822
825 PRINT@288,PC$(1,1):PRINTPC$(
1,2):PRINTUSING"%    %###,###,###
%     %";"==== ";INT(T);" VOTES"
826 PRINTPC$(2,1):PRINTPC$(2,2):
PRINTUSING"%    %###,###,###%
%";"==== ";INT(W);" VOTES"
830 IFT>WTHENMY(1)=MY(1)+YELSEMY
(2)=MY(2)+Y
```

```
834 GOSUB9011:GOSUB9010:CLS:PRIN
T@128,"ELECTORAL VOTE TALLY:"
835 PRINT@192,PC$(1,1):PRINTMY(1
):PRINTPC$(2,1):PRINTMY(2):PRINT
:PRINT" -- 270 VOTES NEEDED TO W
IN --"
840 GOSUB9011:IFMY(1)<270ANDMY(2
)<270THENRETURNELSE850
850 IFFL=1THENRETURN
852 FORPL=1TO8:CLS(PL):SOUNDPL*2
0,2:FORLO=1TO150:NEXTLO:NEXTPL:C
LS:FL=1
860 IFMY(1)>=270THENPL=1ELSEPL=2
870 PRINT@128,"THE WINNER IS . .
 . .              ";PC$(PL,1):PRINT
STRING$(32,".")
890 IFFL=1THEN896ELSE892
892 PRINTPC$(1,1):PRINTMY(1);" E
LECTORAL VOTES":PRINTUSINGH$;X1;
:PRINT" POPULAR VOTES"
893 PRINTPC$(2,1):PRINTMY(2);" E
LECTORAL VOTES":PRINTUSINGH$;Y1;
:PRINT" POPULAR VOTES"
894 PRINT"PRESS <BREAK> TO EXIT"
895 GOTO895
896 PRINT"PRESS <ENTER> TO CONTI
NUE                    THE TALLY":
INPUTX:RETURN
1000 T=INT(VS(LO)/10)
1001 W=VS(LO)-T*10
1002 ONPLGOTO1010,1020
```



```
1010 ONT+1GOTO1011,1013,1015,101
6
1011 VS(LO)=10*CN+W:RETURN
1013 IFCN=1THEN1014ELSEVS(LO)=30
+W
1014 RETURN
1015 IFCN=2THEN1014ELSEVS(LO)=30
+W
1016 RETURN
1020 ONW+1GOTO1021,1023,1025,102
6
1021 VS(LO)=CN+T*10:RETURN
1023 IFCN=1THEN1024ELSEVS(LO)=3+
T*10
1024 RETURN
1025 IFCN=2THEN1024ELSEVS(LO)=3+
T*10
1026 RETURN
2000 PMODE1,1:PCLS2:SCREEN1,0
2010 COLOR3,2
2020 DRAW"BM32,24;L12D10R8L8D10R
12;BM38,18;D20R12;BM72,24;L12D10
R8L8D10R12;BM88,28;L8G4D12F4R8"
2030 DRAW"BM92,24;R12L6D20;BM112
,20;D20;BM124,24;R4F4D12G4L4H4U1
2E4;BM136,16;D20;BM+12,+0;U20;"
2040 LINE(136,16)-(148,36),PSET
2050 DRAW"BM184,20;D4;BM+8,+0;R4
F4D4G4F4D4G4L4H4U4E4R4L4H4U4E4;B
M208,20;D12R12L4U6D14"
2051 DRAW"BM36,64;D16R6E2U4H2L6;
BM+12,+0;D8R8D4G4L2R2E4U12;BM+15
,+0;U8R4F2D4R2D7G3L5U8R7;BM+8,+0
;R8D8L8U8;BM+14,+0;R8D8L8U16"
2052 DRAW"BM128,64;R16L8D16;BM+8
,+0;U8D8R8U8D12G4L2;BM+14,-8;R8U
4L8U4R8;BM+8,+0;R8D8L8U8;BM+12,+
0;D8U4E4R4D8"
2053 COLOR1,2:LINE(0,100)-(255,1
95),PSET,BF
2055 COLOR3,2
2056 DRAW"BM52,152;U26G4U4E8R38D
10R2U6E4R4F4D26F2R2E2F1G4L6H4U8H
6L2D18L8U12L24D12L6"
2057 PAINT(72,132),3,3
2060 COLOR4,2
2062 DRAW"BM148,127;D12R2D12R4U1
2R18D12R5U20R2D11R2U11H4L6D1L17U
1H6U2L1UEL2D4L1G9D4R4E6"
2064 PAINT(158,135),4,4
2070 FORI=1TO15
2072 SCREEN1,1:FORJ=1TO50:NEXT:S
CREEN1,0:FORJ=1TO50:NEXT
2074 SOUNDI*10,1:NEXTI
2080 RETURN
4500 X=RND(15):IFFG(X)=1THEN7690
ELSE4502
4502 FG(X)=1:ONXGOTO4510,4520,45
30,4540,4550,4560,4570,4580,4590
,4600,7690,7690,7690,7690,7690
```

```
4510 PRINT"AP NEWS BULLETIN: LEB
ONESE AND  SYRIAN FORCES EXCHANG
ED GUNFIRE.JEWISH GROUPS REACT I
N NEW YORK AND PENNSYLVANIA - CA
NDIDATES  SHOULD TRAVEL TO THES
E STATES TOMEET WITH JEWISH LEAD
ERS."
4511 PL=3-PL:Y=4:GOSUB4700:Y=5:G
OSUB4700:PL=3-PL:GOTO7690
4520 PRINT"UPI BULLETIN: SUPREME
 COURT     RULES THAT STATES HAV
E THE RIGHTTO SET REASONABLE PEN
ALTIES FOR REFUSING TO BE SEARCH
ED - A.C.L.U. IS OUTRAGED, REPUB
LICAN CANDI-DATE LOSES SUPPORT O
F LIBERAL  GROUPS."
4521 X=PL:PL=1:Y=RND(51)-1:GOSUB
4700:Y=RND(51)-1:GOSUB4700:PL=X:
GOTO7690
4530 PRINT"RUMORS DEVELOP THAT R
EPUBLICAN  CANDIDATE MAY AGREE T
O REMOVE   TOBACCO PRICE SUPPORT
S - TRAVEL TO NORTH CAROLINA(NC)
 IS THE    ONLY WAY TO STOP THE
RUMOR"
4531 X=PL:PL=1:Y=16:GOSUB4700:PL
=X:GOTO7690
4540 PRINT"TIME MAGAZINE HAS END
ORSED THE  REPUBLICAN TICKET BAS
ED ON STA- BILITY OF GOVERNMENT
-  THIS HASRESULTED IN GENERAL S
UPPORT FOR THE REPUBLICAN PARTY"
4541 X=PL:PL=1:FORJ=1TO20:Y=RND(
50)-1:GOSUB4700:NEXTJ
4542 PL=X:GOTO7690
4550 PRINT"AP WIRE STORY: SHUTTL
E LAUNCH   DELAY HAS RESULTED IN
 NEW QUERY ON COST OVERRUNS IN N
ASA - BOTH CANDIDATES SUFFER, RE
PUBLICANS  IN TEXAS AND DEMOCRAT
S IN CALI- FORNIA."
4551 X=PL:PL=1:Y=41:GOSUB4700:PL
=2:Y=42:GOSUB4700:PL=X:GOTO7690
4560 PRINT"NEW YORK TIMES HAS EN
DORSED THE REPUBLICAN PRESIDENTI
AL CANDI-  DATE, AND THE L.A. TI
MES HAS EN-DORSED THE DEMOCRATS.
"
4561 X=PL:FORJ=1TO25:Y=RND(51)-1
:PL=1:GOSUB4700:Y=RND(51)-1:PL=2
:GOSUB4700:NEXTJ
4562 PL=X:GOTO7690
4570 PRINT"SECOND QUARTER ECONOM
IC FIGURES SHOW A SLIGHT INCREAS
E IN NEW   HOUSING STARTS, BUT T
HE COST-OF-LIVING INDEX HAS RISE
N .6% FOR  THE MONTH - DEMOCRATS
 WILL GAIN THE BENEFITS OF THIS
NEWS"
4571 X=PL:PL=1:FORJ=1TO13:Y=RND(
51)-1:GOSUB4700:NEXTJ
4572 PL=X:GOTO7690
4580 PRINT"YOU SHOULD GET TO CHI
CAGO, IL   TO SPEAK TO THE MAYOR
 CONCERNINGHIS REMARKS ABOUT YOU
R FINANCIALDEALINGS."
4581 PL=3-PL:Y=29:GOSUB4700:PL=3
-PL:GOTO7690
4590 PRINT"WALL ST. JOURNAL REPO
RTS THAT   AFTER TAX INCOME HAS
RISEN 1.2% LAST MONTH - REPUBLIC
ANS WILL   BENEFIT THE MOST FROM
 THIS NEWS"
4591 X=PL:PL=2:FORJ=1TO19:Y=RND(
51)-1:GOSUB4700:NEXTJ
4592 PL=X:GOTO7690
4600 PRINT"NATIONAL ENQUIRER REP
ORTS THAT":PRINTPC$(1,2):PRINT"D
EDUCTED EXPENSES ON HIS 1980  I
NCOME TAX FOR LAS VEGAS GAMB-  L
ING LOSSES."
4601 GOSUB9011:PRINT"NOBODY CARE
D...":GOTO7690
4700 VT(Y,PL)=VT(Y,PL)+5:RETURN
5000 CLS:PRINT@128,"GOING OUT ON
 THE CAMPAIGN TRAIL"
5002 K$=S$(CC(PL,CN))
5004 PRINT:PRINT"YOU ARE CURRENT
LY IN ";K$;".":PRINT"WHERE DO YO
```

```
U WANT TO GO?          (ENTER 2-LE
TTER STATE)"
5010 INPUTK$:IFLEN(K$)<>2THEN500
0
5020 X=-1:FORI=0TO50
5030 IFS$(I)=K$THENX=I
5040 NEXTI
5042 IFX<>-1THEN5050
5043 PRINT"THERE IS NO STATE IDE
NTIFIER    LIKE THAT, TRY AGAIN"
:GOSUB9011:GOTO5000
5050 RESTORE:FORI=1TOX:READX1:NE
XT
5060 RESTORE:FORI=1TOCC(PL,CN):R
EADX2:NEXT
5070 RESTORE:FORI=1TO51+X:READY1
:NEXT
5080 RESTORE:FORI=1TO51+CC(PL,CN
):READY2:NEXT
5090 X3=(X1-X2)*(X1-X2)+(Y1-Y2)*
(Y1-Y2)
5100 X3=INT(X3):IFX3<=2THENPRINT
"THIS WILL BE A CHEAP TRIP."ELSE
PRINT"THE PRESS CORPS IS GOING A
LONG."
5110 MY(PL)=MY(PL)-900*X3:PRINT"
THE TRIP COSTS $";900*X3
5120 CC(PL,CN)=X
5130 VT(CC(PL,CN),PL)=VT(CC(PL,C
N),PL)+33-7*CN+RND(8)+2*SR(PL,CN
,3)
5140 GOSUB9011
5150 LO=CC(PL,CN):GOSUB1000
5160 RETURN
5500 CLS:PRINT@128,"ADVERTISING
...":PRINT:PRINT:PRINT"WHERE? (E
NTER 2-LETTER STATE)"
5510 INPUTK$:IFLEN(K$)<>2THEN550
0
5520 X=-1:FORI=0TO50
5530 IFS$(I)=K$THENX=I
5540 NEXTI
5560 IFX=-1THEN5561ELSE5570
5561 PRINT"THERE IS NO STATE IDE
NTIFIER    LIKE THAT, TRY AGAIN"
:GOSUB9011
5562 GOTO5500
5570 RESTORE:FORI=1TO153:READY:N
EXT
5580 JJ=0:FORI=0TOX:READJJ:NEXT
5590 JJ=JJ*50000:PRINT"ADVERTISI
NG IN -";K$;"- COSTS          $";JJ
:GOSUB9010
5600 MY(PL)=MY(PL)-JJ
5610 VT(X,PL)=VT(X,PL)+15+RND(15
)
5620 RETURN
6000 CLS:PRINT@128,"RAISE FUNDS"
:PRINT:PRINT:PRINT"OPTIONS:":PRI
NT"1=RALLY                *HIGH R
ISK2=$100-A-PLATE DINNER *MED  R
ISK3=PRIVATE CONTRIBUTIONS*LOW R
ISK"
6010 PRINT"ENTER 1,2, OR 3":INPU
TX:X=INT(X):IFX<1ORX>3THEN6010EL
SE6020
6020 Y=SR(PL,CN,1)*10000*RND(INT
(10/X)):PRINT"CONGRATULATIONS, Y
OU RAISED IN  EXCESS OF $";Y
6030 VT(CC(PL,CN),PL)=VT(CC(PL,C
N),PL)+20-5*CN*RND(5)
6035 MY(PL)=MY(PL)+Y
6040 GOSUB9011:RETURN
6500 CLS:PRINT@128,"YOU ARE CHAL
LENGING YOUR          OPPONENT TO
A DEBATE...":PRINT
6510 IFRND(3)=2THEN6530ELSE6520
6520 GOSUB9010:PRINT:PRINT"HE DE
CLINES . . . BUT YOU PICK  UP SU
PPORT NATIONWIDE FOR YOUR  INITI
ATIVE":GOSUB9011
6521 FORI=1TO25:X=RND(51)-1:VT(X
,PL)=VT(X,PL)+RND(5):NEXTI
6522 RETURN
6530 GOSUB9011:PRINT"HE ACCEPTS
... BUT WILL ONLY    DEBATE YOU
ON";
6532 X=RND(4)
6534 ONXGOTO6535,6536,6537
```

```
6535 PRINT" DEFENSE":GOTO6540
6536 PRINT" THE ECONOMY":GOTO654
0
6537 PRINT" SOCIAL PROGRAMS":GOT
O6540
6538 PRINT" HIS TURF":GOTO6540
6540 PRINT:GOSUB9010:X=RND(2):IF
X=1THENPRINT"THIS HITS YOUR WEAK
NESS..."ELSEPRINT"THIS FITS YOUR
 STRATEGY..."
6542 PRINT:PRINT"DO YOU STILL WA
NT TO DEBATE? Y/N"
6544 K$=INKEY$:IFK$=""THEN6544EL
SEIFK$="Y"THEN6546ELSEIFK$="N"TH
EN6548ELSE6544
6546 GOSUB9011:X=3-PL
6547 Y=SR(PL,CN,2)+SR(PL,CN,3)-S
R(X,CN,2)-SR(X,CN,3)+3-RND(6):GO
TO6560
6548 GOSUB9011:PRINT" OK, BUT YO
U LOSE SOME RESPECT  OF THE VOTE
RS":GOSUB9011
6549 IFPL=1THENX=2ELSEX=1
6550 FORI=1TO20:Y=RND(51)-1:VT(Y
,X)=VT(Y,X)+RND(3)
6551 GOSUB9010:RETURN
6560 IFABS(Y)<1THEN6570ELSEIFY>0
THEN6580ELSE6590
6570 PRINT"THE DEBATE WAS A VIRT
UAL DRAW":GOSUB9011:RETURN
6580 PRINT"YOU WERE A CLEAR CUT
WINNER OF  THE DEBATE":GOSUB9011
6582 FORI=0TO50:VT(I,PL)=VT(I,PL
)+RND(4):NEXT
6584 RETURN
6590 PRINT"YOU STUMBLED ON SOME
IMPORTANT  POINTS ... THE PRESS
FEELS THAT YOU LOST THE DEBATE"
6592 IFPL=1THENX=2ELSEX=1
6594 FORI=0TO50:VT(I,X)=VT(I,X)+
RND(4):NEXT
6595 GOSUB9010:GOSUB9010
6596 RETURN
7000 CLS:PRINT@128,"YOU ARE YIEL
DING YOUR POSITION  ...YOU WILL
GAIN STRENGTH IN    THIS STATE"
7010 GOSUB9011
7020 VT(CC(PL,CN),PL)=VT(CC(PL,C
N),PL)+20-5*CN+RND(5)
7025 MY(PL)=MY(PL)-1000
7030 RETURN
7500 RESTORE:FORJ=1TO204:READX:N
EXT
7510 CLS:FORI=0TO50
7520 READC$(1)
7530 T=INT(VS(I)/10):W=VS(I)-T*1
0
7532 IFT=1THENX=188ELSEIFT=2THEN
X=179ELSEIFT=3THENX=191ELSEX=143
7534 IFW=1THENY=172ELSEIFW=2THEN
Y=163ELSEIFW=3THENY=175ELSEY=143
7540 PRINT@I*8,S$(I)+"-"+C$(1)+"
 ";CHR$(X);CHR$(Y);
7550 NEXTI
7560 PRINT:PRINT"PRESS <ENTER> F
OR MORE INFO"
7570 K$=INKEY$:IFK$=CHR$(13)THEN
7580ELSE7570
7580 CLS:X=0:Y=0
7590 FORI=0TO50:X=X+VT(I,1):Y=Y+
VT(I,2):NEXT
7600 PRINT"GALLUP POLL SHOWS:"
7610 PRINTPC$(1,1):PRINTTAB(14),
INT(100*X/(X+Y));"%"
7620 PRINTPC$(2,1):PRINTTAB(14),
INT(100*Y/(X+Y));"%"
7630 GOTO4500
7690 PRINT"PRESS <ENTER> TO CONT
INUE"
7695 K$=INKEY$:IFK$=CHR$(13)THEN
7800ELSE7695
7800 ONPLGOTO7801,7803
7801 CLS4:PRINT@10,"DEMOCRATIC";
7802 GOTO7804
7803 CLS3:PRINT@10,"REPUBLICAN";
7804 ONCNGOTO7805,7807
7805 PRINT@36," PRESIDENTIAL CAN
DIDATE "
7806 GOTO7808
```

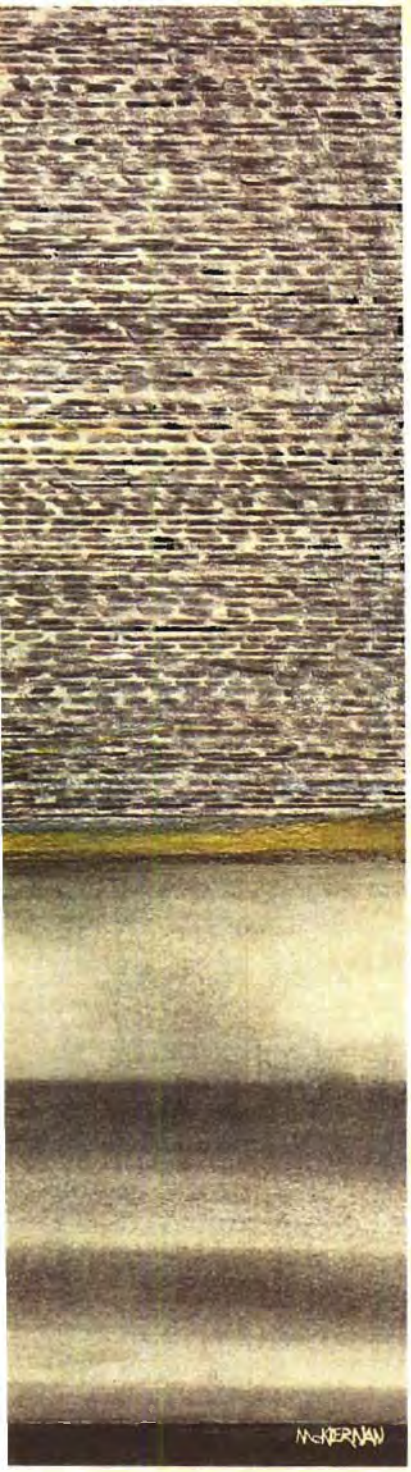
```
7807 PRINT@32," VICE-PRESIDENTIA
L CANDIDATE "
7808 PRINTPC$(PL,CN)
7809 IFDY>1THENPRINTCHR$(255);"
";DY;"DAYS UNTIL THE ELECTION  "
;CHR$(255);ELSEPRINT" ";CHR$(255
);CHR$(255);" TOMORROW'S THE ELE
CTION! ";CHR$(255);CHR$(255);" "
;
7810 PRINT@129,"FINANCIAL STRENG
TH";SR(PL,CN,1)
7812 PRINT@161,"EXPERIENCE
  ";SR(PL,CN,2)
7813 PRINT@193,"CHARISMA
  ";SR(PL,CN,3)
7814 IF(MD-10=PL)THEN5000
7815 IFMY(PL)>00RDY<3THEN7816ELS
E7838
7816 PRINTUSING"%
 %$$,###,###";"MONEY AVAILABLE
",MY(PL)
7817 PRINT"     YOU ARE CURRENTLY
 IN -";S$(CC(PL,CN));"-      "
7818 PRINT@321,"T=TRAVEL
A=ADVERTISE"
7820 PRINT@353,"R=RAISE FUNDS
D=DEBATE"
7822 PRINT@385,"Y=YIELD
I=INTELLIGENCE":SOUND100,5
7830 K$=INKEY$:IFK$=""THEN7830
7832 IFK$="T"ORK$="A"ORK$="R"ORK
$="D"ORK$="Y"THENRETURNELSE7834
7834 IFK$="I"THEN7500ELSE7830
7838 PRINTUSING"%      %$$,###,##
#";"YOU OWE ",ABS(MY(PL)):PRINT"
YOU MUST RAISE FUNDS":GOSUB9011:
K$="R":GOTO7832
8000 CLS:PRINT"DIRECTIONS FOR CA
NDIDATE          SELECTION:
                 CHOOSE PRESIDENTI
AL AND VICE      PRESIDENTIAL CAND
IDATES FROM THE";
8001 PRINT"LIST. STRENGTHS ARE C
ODED BY     F=FINANCIAL(1-5)
            E=EXPERIENCE(1-5)
            C=CHARISMA(1-5)"
8003 PRINT:PRINT:PRINT"HIT ENTER
 TO CONTINUE"
8004 K$=INKEY$:IFK$=CHR$(13)THEN
8005ELSE8004
8005 RETURN
8010 CLS:C$(1)="SEN. JOHN GLENN
       2 1 4":CS$(1)="OH":C$(2)=
"SEN. ALAN CRANSTON      3 2 1":C
S$(2)="CA":C$(3)="GOV. JERRY BRO
WN         1 2 4":CS$(3)="CA"
8011 C$(4)="SEN. WALTER MONDALE
   2 4 2":CS$(4)="MN":C$(5)="GOV
```

```
. REUBEN ASKEW        1 1 2":CS$(5
)="FL":C$(6)="SEN. GARY HART
      1 1 3":CS$(6)="CO"
8012 FLAG=●:PRINT"SELECT PRESIDE
NTIAL CANDIDATE        DEMOCRATIC
PARTY":GOSUB8090
8013 INPUTFL:FL=INT(FL):IFFL<1OR
FL>6THEN8013ELSE8014
8014 CLS:PRINT"SELECT RUNNING MA
TE":GOSUB8090
8015 INPUTX:X=INT(X):IFX<1ORX>60
RX=FLTHEN8015ELSE8016
8016 GOSUB8060
8017 PC$(1,2)=PC$(1,2)+" D-"+CS$
(X):PC$(1,1)=PC$(1,1)+" D-"+CS$(
FL)
8018 RETURN
8050 CLS:C$(1)="PRES. RONALD REA
GAN     3 5 3":CS$(1)="CA":C$(2)=
"VICE-PRES. GEORGE BUSH 1 3 2":C
S$(2)="MI":C$(3)="SEN. LOWELL WE
ICKER     1 2 4":CS$(3)="CT"
8051 C$(4)="SEN. ROBERT DOLE
   3 2 1":CS$(4)="KS":C$(5)="SEN
. HOWARD BAKER        1 3 3":CS$(5
)="TN"
8052 C$(6)="SEN. PAUL LAXALT
   2 1 2":CS$(6)="NV":CLS:PRINT"
SELECT PRESIDENTIAL CANDIDATE
   REPUBLICAN":FLAG=0:GOSUB8090
8053 INPUTFL:FL=INT(FL):IFFL<1OR
FL>6THEN8053ELSE8054
8●54 CLS:PRINT"SELECT RUNNING MA
TE":GOSUB8090
8055 INPUTX:X=INT(X):IFX<1ORX>60
RX=FLTHEN8055ELSE8056
8056 GOSUB8060
8057 PC$(2,2)=PC$(2,2)+" R-"+CS$
(X):PC$(2,1)=PC$(2,1)+" R-"+CS$(
FL)
8058 RETURN
8060 PC$(J,1)=LEFT$(C$(FL),23):P
C$(J,2)=LEFT$(C$(X),23)
8061 PRINT"THE STATE CHAIRMEN AR
E GIVEN     THEIR INSTRUCTIONS...
"
8062 FORI=1TO3
8063 SR(J,1,I)=VAL(MID$(C$(FL),2
2+2*I,1))
8064 SR(J,2,I)=VAL(MID$(C$(X),22
+2*I,1))
8065 NEXTI
8067 HS$(J,1)=CS$(FL):HS$(J,2)=C
S$(X)
8069 FORI=0TO50
8070 IFHS$(J,1)=S$(I)THENCC(J,1)
=I
8071 IFHS$(J,2)=S$(I)THENCC(J,2)
=I
8072 NEXT
8076 CN=1:PL=J:LO=CC(J,1):GOSUB1
000:VT(LO,PL)=10+VT(LO,PL)
8077 CN=2:PL=J:LO=CC(J,2):GOSUB1
000:VT(LO,PL)=VT(LO,PL)+10
8078 VT(LO,PL)=VT(LO,PL)+10
8079 RETURN
8090 PRINT" # HOPEFULS
     F E C":FORJJ=1TO6
8091 IFFLAG=JJTHEN8095
8092 PRINTJJ;C$(JJ):PRINT"     "+
CS$(JJ)
8095 NEXT
8096 RETURN
8959 REM LOCATIONS X,X,Y,Y
8960 DATA 12,11,12,12,11,11,13,1
2,12,11,12,11,8,9,10,11,11,10,9,
9,10,10,9,8
8961 DATA 9,8,9,9,10,9,8,8,7,7,7
,6,6,4,5,2,2,7,2,4,5,3,4,1,5,3,1

8962 DATA 2,3,2,2,2,2,1,2,2,3,2,
3,4,3,4,3,3,5,4,3,4,3,4,4
8963 DATA 2,3,2,2,2,3,1,2,3,3,2,
1,2,4,3,2,1,4,3,3,4,2,1,5,2,3,1
8969 REM EDGES
8970 DATA 0,0,0,1,0,0,1,0,1,1,1,
0,0,1,0,1,0,1,0,0,0,0,0,0
8971 DATA 1,0,1,0,1,1,0,1,1,1,1,
1,1,1,1,1,0,0,1,1,1,1,1,0,1,1,1
8979 REM ADV COSTS
8980 DATA 1,1,2,2,3,4,1,1,1,1,1,
1,1,1,2,2,2,2,1,1,1,1,1,1
8981 DATA 2,2,2,3,3,3,1,1,1,1,1,
1,1,1,1,1,1,3,4,1,1,1,1,1,1,1,1
8989 REM ELEC VOTES
8990 DATA 8,10,14,17,27,41,4,4,4
,3,3,3,10,10,12,12,13,17,9,9,8,6
,7,6
8991 DATA 11,12,13,21,25,26,10,8
,8,7,5,3,4,6,7,6,9,26,45,4,4,4,4
,4,3,3,3
8999 REM STATES AND AVG VOTES
9000 DATA CT,MD,MA,NJ,PA,NY,ME,R
I,NH,DE,VT,DC,LA,TN,GA,VA,NC,FL,
AL,KY,SC,WV,MS,AR
9001 DATA WI,MO,IN,MI,OH,IL,MN,I
A,OK,KS,NE,ND,SD,AZ,CO,OR,WA,TX,
CA,UT,NM,ID,MT,HI,WY,NV,AK
9002 DATA 150,150,250,300,450,65
0,40,40,30,23,18,13,130,150,140,
170,170,300,120,110,80,8,70,70
9003 DATA 200,200,210,360,400,50
0,180,120,100,90,60,30,30,70,100
,100,150,400,750,50,40,32,32,30,
15,20,11
9010 FORDL=1TO1200:NEXT:RETURN
9011 FORDL=1TO2400:NEXT:RETURN
10000 CLEAR500:PCLEAR2
10010 GOTO10
```

WAR

GAME



by Thomas G. Weber

The idea for this simulation came from the movie "WarGames" which was about a high school student who taps into the United States defense computer, and by doing so, almost starts World War III.

In this simulation, you are in control of the NORth American Defense (NORAD) facility, and are required to take part in a simulation of a thermonuclear war between the U.S. and Russia. Both countries have 10 bases. The country to lose all 10 of its bases first loses the simulation. No matter which side loses first, though, chances are that the other side wouldn't do much better, due to the damage done to its country.

In *WarGame,* you'll have eight commands you can enter: USCON, SOVCON, JAM, SUBS?, DESUB, LAUNCH, TIME? and ?, which prints out the previous list. Every command except ? must be ended with a carriage return in order to be acted upon.

USCON prints out the condition of the United States in terms of estimated kill ratio, and bases remaining.

SOVCON does the same thing as USCON except using data concerning the Soviet Union.

JAM will jam Russia's launch code for a short time, during which they cannot launch any missiles from their bases, other than one which may already be in the air (it should be noted that Russia can do the same thing to you).

SUBS? will detect any enemy subs near the Pacific coast.

DESUB is a special mode you enter to destroy these subs. It is important that this be done, since these vessels launch missiles at your bases as frequently as, or more so than, the Soviet bases do. Even though knocking out the subs will save your bases from destruction, it will not affect the number of bases you have to get. Along the same lines, you don't have to get any of the subs to conclude the simulation . . . just the bases. When you enter the sub destroy mode, a flashing dot will appear in the center of the larger map of the United States. You can move the dot around anywhere within the boundary of this map. The idea is to position it near the center of a submarine using the four arrow keys. When you hit the "kill spot" on a sub, you get a pulsing tone, telling you to launch the missile (by hitting [ENTER]). Make sure that before you launch, you hear the tone at least twice *with your fingers off the arrow keys.* This assures that the sub will really be destroyed, rather than a near-miss. Once you've launched a missile in this mode, your flashing dot will return to where it started. You can now either go get another sub, or return to the regular com-

mand mode by hitting [BREAK]. You can also return at any time while in the DESUB mode by hitting [BREAK].

Launch—the syntax for this command is "LAUNCHx-y," where x is the number corresponding to the U.S. base launch site, and y is the number corresponding to the Soviet base target. (Don't forget the "-" in between them.)

TIME? will tell you, in real time, the "game time elapsed" and the "estimated time remaining" in the simulation.

Not all the missiles launched by one or the other side will make it all the way to their destinations. Occasionally, a missile may fail to launch at the launch site or malfunction in midflight. Neither of these indicate that the launch site base will not launch future warheads correctly, but rather indicate a fault with the individual missile.

Usually this simulation is completed within five to 10 minutes, so think and act quickly; and Good Luck, General . . .

Here are the names of all the bases, along with the numbers corresponding to them:

Soviet Bases:	U.S. Bases:
1) Moscow	1) Nome
2) Leningrad	2) Fairbanks
3) Kharkov	3) San Francisco
4) Volograd	4) San Diego
5) Rostov	5) Spokane
6) Archangel	6) Denver
7) Yakutsk	7) Houston
8) Kamchatka	8) Charleston
9) Vladivostok	9) Chicago
10) Minsk	10) Seattle

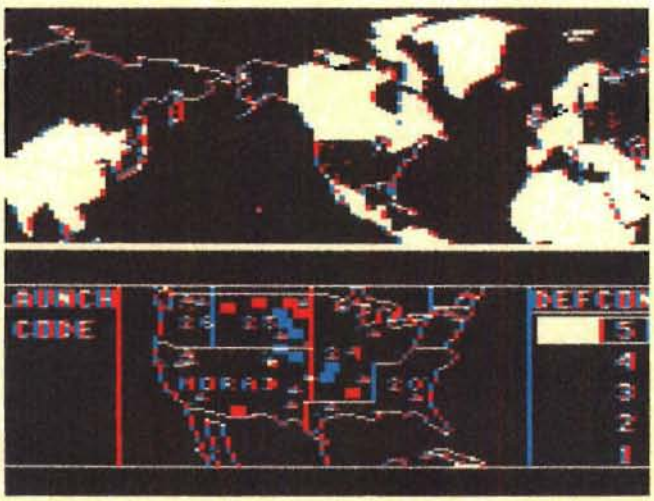
Loading Procedures

Using an Editor/Assembler like Radio Shack's *EDTASM+*, type in the program called PART 1. This routine is similar to that used in the "Micro-Meltdown" article (*Rainbow*, April 1983), to allow the player to type in Hi-Res, but contains a different character-set, and contains a few other ML routines.

PART 2, which draws the NORAD war room scene, takes a long time to load and run, so for this reason, after completing its picture, it saves it out to tape as a ML file which is then loaded back in with a *CLOADM* command. The picture loads and is ready to go much faster than it took to load and run PART 2, so after you have saved its ML file, you can kill the BASIC program if desired.

PART 3 is the instruction program. It gives a brief description of what you must do, and also will help you to match the base numbers to their locations. This program will automatically load in the next one.

PART 4 is the actual simulation part of the program, and, like PART 2, takes awhile to load. When PART 3 has finished loading PART 4, you'll see an "OK" prompt in the lower left corner of the screen. You can type *RUN* and [ENTER], then the war is on . . .



Editor's Note: *WarGame* uses a speed-up poke in Lines 40 and 101. If your computer will not accept the speed-up poke, remove POKE 65495,0 from these lines.)

(Thomas Weber is a freshman at Marquette University, majoring in Computer Science. He is an active member of a CoCo Club in Milwaukee, Wis.)

Listing 1

```
                        00010 * PART 1 SOURCE CODE *
5120                    00020         ORG $5120
                        00025 * ROUTINE TO ALLOW SCREEN OUTPUT IN HI-RES *
5120 6D  8C 31          00030         TST $5154,PCR
5123 26  18             00040         BNE $5140
5125 6C  8C 2C          00050         INC $5154,PCR
5128 BE  0168           00060         LDX $168        POINTER TO 'CONSOLE OUT'
5128 AF  8C 27          00070         STX $5155,PCR
512E 30  8C 68          00080         LEAX $5199,PCR
5131 BF  0168           00090         STX $168
5134 BE  019B           00100         LDX $19B        POINTER TO BASIC'S COMMAND
5137 AF  8C 1E          00110         STX $5158,PCR                INTERPRETATION LOOP
513A 30  8C 1D          00120         LEAX $515A,PCR
513D BF  0198           00130         STX $198
5140 39                 00140         RTS
                        00145 * ROUTINE TO SWITCH SCREEN OUTPUT BACK TO LO-RES ONLY *
5141 6D  8C 10          00150         TST $5154,PCR
5144 27  FA             00160         BEQ $5140
5146 6F  8C 0B          00170         CLR $5154,PCR
5149 AE  8C 09          00180         LDX $5155,PCR
514C BF  0168           00190         STX $168
514F AE  8C 06          00200         LDX $5158,PCR
5152 20  E9             00210         BRA $513D
                        00215 * CONSOLE OUT DATA *
5154     00             00220         FCB $00
5155     82             00230         FCB $82
5156     73             00240         FCB $73
5157     00             00250         FCB $00
5158     82             00260         FCB $82
5159     B9             00270         FCB $B9
                        00275 * NEW 'COMMAND INTERPRETATION LOOP' FOR BASIC *
515A 34  07             00280         PSHS CC,A,B
515C F6  0155           00290         LDB $155
515F B6  F7             00300         LDA $F7
5161 B7  FF02           00310         STA $FF02
5164 B6  FF00           00320         LDA $FF00
5167 8A  80             00330         ORA #$80
5169 81  F7             00340         CMPA #$F7
516B 27  04             00350         BEQ $5171
516D CA  08             00360         ORB #$08
516F 20  02             00370         BRA $5173
5171 C4  F7             00380         ANDB #$F7
5173 F7  0155           00390         STB $155
5176 F6  0156           00400         LDB $156
5179 86  EF             00410         LDA #$EF
517B B7  FF02           00420         STA $FF02
517E B6  FF00           00430         LDA $FF00
5181 8A  80             00440         ORA #$80
5183 81  F7             00450         CMPA #$F7
5185 27  04             00460         BEQ $518B
5187 CA  08             00470         ORB #$08
5189 20  02             00480         BRA $518D
518B C4  F7             00490         ANDB #$F7
518D F7  0156           00500         STB $156
5190 35  07             00510         PULS CC,A,B
5192 32  62             00520         LEAS $02,S
5194 1C  AF             00530         ANDCC #$AF
5196 7E  ADA5           00540         JMP $ADA5
                        00545 * NEW 'CONSOLE OUT' ROUTINE FOR BASIC *
5199 34  36             00550         PSHS A,B,X,Y
519B 81  08             00560         CMPA #$08       CHECK FOR BACKSPACE
519D 1027 0099          00570         LBEQ $523A
51A1 20  0C             00580         BRA $51AF
51AF                    00590         ORB #$1AF
51AF C6  2D             00600         LDB #$2D        CHECK FOR VALID KEYPRESS
51B1 31  8D 0093        00610         LEAY $5248,PCR
51B5 A1  A0             00620         CMPA ,Y+
51B7 27  30             00630         BEQ $51E9
51B9 31  27             00640         LEAY 7,Y
51BB 5A                 00650         DECB
51BC 26  F7             00660         BNE $51B5
51BE C6  24             00670         LDB #$24
51C0 A1  A0             00680         CMPA ,Y+        CHECK FOR DESCENDER
```

```
51C2 27  07     00690        BEQ $51CB
51C4 31  25     00700        LEAY 5,Y
51C6 5A         00710        DECB
51C7 26  F7     00720        BNE $51C0
51C9 35  B6     00730        PULS PC,Y,X,B,A
51CB BD  22     00740        BSR $51EF       GET HI-RES PRINT POSITION
51CD C6  01     00750        LDB #$01
51CF 12         00760        NOP
51D0 E7  8C 39  00770        STB $520C,PCR
51D3 C6  39     00780        LDB #$39
51D5 12         00790        NOP
51D6 E7  8C 5B  00800        STB $5234,PCR
51D9 17  02AD   00810        LBSR $5489
51DC 12         00820        NOP
51DD 12         00830        NOP
51DE E7  8C 2B  00840        STB $520C,PCR
51E1 C6  A7     00850        LDB #$A7
51E3 12         00860        NOP
51E4 E7  8C 4D  00870        STB $5234,PCR
51E7 20  E0     00880        BRA $51C9
51E9 BD  04     00890        BSR $51EF
51EB BD  1E     00900        BSR $520B
51ED 20  DA     00910        BRA $51C9
                00915 * TRANSLATE PRINT POSITION FROM LO TO HI-RES *
51EF DC  88     00920        LDD #$88 GET LO-RES SCREEN PRINT POSITION
51F1 34  04     00930        PSHS B
51F3 44         00940        LSRA
51F4 56         00950        RORB
51F5 54         00960        LSRB
51F6 54         00970        LSRB
51F7 54         00980        LSRB
51F8 54         00990        LSRB
51F9 86  0C     01000        LDA #$0C
51FB 3D         01010        MUL
51FC 86  20     01020        LDA #$20
51FE 3D         01030        MUL
51FF 1F  01     01040        TFR D,X
5201 35  04     01050        PULS B
5203 C4  1F     01060        ANDB #$1F
5205 3A         01070        ABX
5206 DC  BC     01080        LDD #$BC
5208 30  8B     01090        LEAX D,X
520A 39         01100        RTS
                01105 * PRINT CHARACTER IN HI-RES *
520B C6  03     01110        LDB #$03
520D 4F         01120        CLRA
520E 20  07     01130        BRA $5217
5217            01140        ORG $5217
5217 A7  00     01150        STA 0,X CLEAR AREA ABOVE CHARACTER
5219 30  88 20  01160        LEAX $20,X
521C 5A         01170        DECB
521D 26  FB     01180        BNE $5217
521F C6  07     01190        LDB #$07
5221 A6  A0     01200        LDA ,Y+
5223 20  05     01210        BRA $522A
522A            01220        ORG $522A
522A A7  00     01230        STA 0,X CLEAR AREA BELOW CHARACTER
522C 30  88 20  01240        LEAX $20,X
522F 5A         01250        DECB
5230 26  EF     01260        BNE $5221
5232 86  00     01270        LDA #$00
5234 A7  00     01280        STA $0,X
5236 A7  88 20  01290        STA $20,X
5239 39         01300        RTS
                01305 * BACKSPACE ROUTINE *
523A 17  FFB2   01310        LBSR $51EF
523D 30  1F     01320        LEAX -1,X
523F 31  8D 0126 01330       LEAY $5369,PCR
5243 BD  C6     01340        BSR $520B
5245 16  FFB1   01350        LBRA $51C9
                01355 * START OF CHARACTER DATA *
5248    41      01360        FCB $41
5249    18      01370        FCB $18
524A    3C      01380        FCB $3C
524B    66      01390        FCB $66
524C    66      01400        FCB $66
524D    7E      01410        FCB $7E
524E    66      01420        FCB $66
524F    66      01430        FCB $66
5250    42      01440        FCB $42
5251    7C      01450        FCB $7C
5252    66      01460        FCB $66
5253    66      01470        FCB $66
5254    7C      01480        FCB $7C
5255    66      01490        FCB $66
5256    66      01500        FCB $66
5257    7C      01510        FCB $7C
5258    43      01520        FCB $43
5259    3C      01530        FCB $3C
525A    66      01540        FCB $66
525B    60      01550        FCB $60
525C    60      01560        FCB $60
525D    60      01570        FCB $60
525E    66      01580        FCB $66
525F    3C      01590        FCB $3C
5260    44      01600        FCB $44
5261    78      01610        FCB $78
5262    6C      01620        FCB $6C
5263    66      01630        FCB $66
5264    66      01640        FCB $66
5265    66      01650        FCB $66
5266    6C      01660        FCB $6C
5267    78      01670        FCB $78
5268    45      01680        FCB $45
5269    7E      01690        FCB $7E
526A    60      01700        FCB $60
526B    60      01710        FCB $60
```

```
526C  7C  01720  FCB $7C
526D  60  01730  FCB $60
526E  60  01740  FCB $60
526F  7E  01750  FCB $7E
5270  46  01760  FCB $46
5271  7E  01770  FCB $7E
5272  60  01780  FCB $60
5273  60  01790  FCB $60
5274  7C  01800  FCB $7C
5275  60  01810  FCB $60
5276  60  01820  FCB $60
5277  60  01830  FCB $60
5278  47  01840  FCB $47
5279  3C  01850  FCB $3C
527A  66  01860  FCB $66
527B  60  01870  FCB $60
527C  60  01880  FCB $60
527D  6E  01890  FCB $6E
527E  66  01900  FCB $66
527F  3C  01910  FCB $3C
5280  48  01920  FCB $48
5281  66  01930  FCB $66
5282  66  01940  FCB $66
5283  66  01950  FCB $66
5284  7E  01960  FCB $7E
5285  66  01970  FCB $66
5286  66  01980  FCB $66
5287  66  01990  FCB $66
5288  49  02000  FCB $49
5289  3C  02010  FCB $3C
528A  18  02020  FCB $18
528B  18  02030  FCB $18
528C  18  02040  FCB $18
528D  18  02050  FCB $18
528E  18  02060  FCB $18
528F  3C  02070  FCB $3C
5290  4A  02080  FCB $4A
5291  06  02090  FCB $06
5292  06  02100  FCB $06
5293  06  02110  FCB $06
5294  06  02120  FCB $06
5295  06  02130  FCB $06
5296  66  02140  FCB $66
5297  3C  02150  FCB $3C
5298  4B  02160  FCB $4B
5299  66  02170  FCB $66
529A  66  02180  FCB $66
529B  6C  02190  FCB $6C
529C  78  02200  FCB $78
529D  6C  02210  FCB $6C
529E  66  02220  FCB $66
529F  66  02230  FCB $66
52A0  4C  02240  FCB $4C
52A1  60  02250  FCB $60
52A2  60  02260  FCB $60
52A3  60  02270  FCB $60
52A4  60  02280  FCB $60
52A5  60  02290  FCB $60
52A6  60  02300  FCB $60
52A7  7E  02310  FCB $7E
52A8  4D  02320  FCB $4D
52A9  66  02330  FCB $66
52AA  7E  02340  FCB $7E
52AB  7E  02350  FCB $7E
52AC  66  02360  FCB $66
52AD  66  02370  FCB $66
52AE  66  02380  FCB $66
52AF  66  02390  FCB $66
52B0  4E  02400  FCB $4E
52B1  66  02410  FCB $66
52B2  76  02420  FCB $76
52B3  7E  02430  FCB $7E
52B4  7E  02440  FCB $7E
52B5  66  02450  FCB $66
52B6  66  02460  FCB $66
52B7  66  02470  FCB $66
52B8  4F  02480  FCB $4F
52B9  7E  02490  FCB $7E
52BA  66  02500  FCB $66
52BB  66  02510  FCB $66
52BC  66  02520  FCB $66
52BD  66  02530  FCB $66
52BE  66  02540  FCB $66
52BF  7E  02550  FCB $7E
52C0  50  02560  FCB $50
52C1  7C  02570  FCB $7C
52C2  66  02580  FCB $66
52C3  66  02590  FCB $66
52C4  7C  02600  FCB $7C
52C5  60  02610  FCB $60
52C6  60  02620  FCB $60
52C7  60  02630  FCB $60
52C8  51  02640  FCB $51
52C9  3C  02650  FCB $3C
52CA  66  02660  FCB $66
52CB  66  02670  FCB $66
52CC  66  02680  FCB $66
52CD  76  02690  FCB $76
52CE  6C  02700  FCB $6C
52CF  3A  02710  FCB $3A
52D0  52  02720  FCB $52
52D1  7C  02730  FCB $7C
52D2  66  02740  FCB $66
52D3  66  02750  FCB $66
52D4  7C  02760  FCB $7C
52D5  6C  02770  FCB $6C
52D6  66  02780  FCB $66
52D7  66  02790  FCB $66
52D8  53  02800  FCB $53
52D9  3C  02810  FCB $3C
52DA  66  02820  FCB $66
52DB  60  02830  FCB $60
52DC  3C  02840  FCB $3C
52DD  06  02850  FCB $06
52DE  66  02860  FCB $66
52DF  3C  02870  FCB $3C
52E0  54  02880  FCB $54
52E1  7E  02890  FCB $7E
52E2  18  02900  FCB $18
52E3  18  02910  FCB $18
52E4  18  02920  FCB $18
52E5  18  02930  FCB $18
52E6  18  02940  FCB $18
52E7  18  02950  FCB $18
52E8  55  02960  FCB $55
52E9  66  02970  FCB $66
52EA  66  02980  FCB $66
52EB  66  02990  FCB $66
52EC  66  03000  FCB $66
52ED  66  03010  FCB $66
52EE  66  03020  FCB $66
52EF  3C  03030  FCB $3C
52F0  56  03040  FCB $56
52F1  66  03050  FCB $66
52F2  66  03060  FCB $66
52F3  66  03070  FCB $66
52F4  66  03080  FCB $66
52F5  66  03090  FCB $66
52F6  3C  03100  FCB $3C
52F7  18  03110  FCB $18
52F8  57  03120  FCB $57
52F9  66  03130  FCB $66
52FA  66  03140  FCB $66
52FB  66  03150  FCB $66
52FC  66  03160  FCB $66
52FD  7E  03170  FCB $7E
52FE  7E  03180  FCB $7E
52FF  66  03190  FCB $66
5300  58  03200  FCB $58
5301  66  03210  FCB $66
5302  66  03220  FCB $66
5303  3C  03230  FCB $3C
5304  18  03240  FCB $18
5305  3C  03250  FCB $3C
5306  66  03260  FCB $66
5307  66  03270  FCB $66
5308  59  03280  FCB $59
5309  66  03290  FCB $66
530A  66  03300  FCB $66
530B  3C  03310  FCB $3C
530C  18  03320  FCB $18
530D  18  03330  FCB $18
530E  18  03340  FCB $18
530F  18  03350  FCB $18
5310  5A  03360  FCB $5A
5311  7E  03370  FCB $7E
5312  06  03380  FCB $06
5313  0C  03390  FCB $0C
5314  18  03400  FCB $18
5315  30  03410  FCB $30
5316  60  03420  FCB $60
5317  7E  03430  FCB $7E
5318  30  03440  FCB $30
5319  3C  03450  FCB $3C
531A  66  03460  FCB $66
5318  6E  03470  FCB $6E
531C  7E  03480  FCB $7E
531D  76  03490  FCB $76
531E  66  03500  FCB $66
531F  3C  03510  FCB $3C
```

```
5320 31 03520 FCB $31
5321 18 03530 FCB $18
5322 38 03540 FCB $38
5323 18 03550 FCB $18
5324 18 03560 FCB $18
5325 18 03570 FCB $18
5326 18 03580 FCB $18
5327 3C 03590 FCB $3C
5328 32 03600 FCB $32
5329 3C 03610 FCB $3C
532A 66 03620 FCB $66
532B 06 03630 FCB $06
532C 0C 03640 FCB $0C
532D 18 03650 FCB $18
532E 30 03660 FCB $30
532F 7E 03670 FCB $7E
5330 33 03680 FCB $33
5331 3C 03690 FCB $3C
5332 66 03700 FCB $66
5333 06 03710 FCB $06
5334 1C 03720 FCB $1C
5335 06 03730 FCB $06
5336 66 03740 FCB $66
5337 3C 03750 FCB $3C
5338 34 03760 FCB $34
5339 0E 03770 FCB $0E
533A 1E 03780 FCB $1E
533B 36 03790 FCB $36
533C 66 03800 FCB $66
533D 7E 03810 FCB $7E
533E 06 03820 FCB $06
533F 06 03830 FCB $06
5340 35 03840 FCB $35
5341 7E 03850 FCB $7E
5342 60 03860 FCB $60
5343 7C 03870 FCB $7C
5344 06 03880 FCB $06
5345 06 03890 FCB $06
5346 66 03900 FCB $66
5347 3C 03910 FCB $3C
5348 36 03920 FCB $36
5349 3C 03930 FCB $3C
534A 66 03940 FCB $66
5348 60 03950 FCB $60
534C 7C 03960 FCB $7C
534D 66 03970 FCB $66
534E 66 03980 FCB $66
534F 3C 03990 FCB $3C
5350 37 04000 FCB $37
5351 7E 04010 FCB $7E
5352 06 04020 FCB $06
5353 06 04030 FCB $06
5354 0C 04040 FCB $0C
5355 18 04050 FCB $18
5356 30 04060 FCB $30
5357 60 04070 FCB $60
5358 38 04080 FCB $38
5359 3C 04090 FCB $3C
535A 66 04100 FCB $66
535B 66 04110 FCB $66
535C 3C 04120 FCB $3C
535D 66 04130 FCB $66
535E 66 04140 FCB $66
535F 3C 04150 FCB $3C
5360 39 04160 FCB $39
5361 3C 04170 FCB $3C
5362 66 04180 FCB $66
5363 66 04190 FCB $66
5364 3E 04200 FCB $3E
5365 06 04210 FCB $06
5366 66 04220 FCB $66
5367 3C 04230 FCB $3C
5368 20 04240 FCB $20
5369 00 04250 FCB $00
536A 00 04260 FCB $00
536B 00 04270 FCB $00
536C 00 04280 FCB $00
536D 00 04290 FCB $00
536E 00 04300 FCB $00
536F 00 04310 FCB $00
5370 25 04320 FCB $25
5371 30 04330 FCB $30
5372 4A 04340 FCB $4A
5373 34 04350 FCB $34
5374 08 04360 FCB $08
5375 2C 04370 FCB $2C
5376 52 04380 FCB $52
5377 0C 04390 FCB $0C
5378 27 04400 FCB $27
5379 18 04410 FCB $18
537A 18 04420 FCB $18
537B 18 04430 FCB $18
537C 00 04440 FCB $00
537D 00 04450 FCB $00
537E 00 04460 FCB $00
537F 00 04470 FCB $00
5380 2A 04480 FCB $2A
5381 10 04490 FCB $10
5382 54 04500 FCB $54
5383 38 04510 FCB $38
5384 38 04520 FCB $38
5385 54 04530 FCB $54
5386 10 04540 FCB $10
5387 00 04550 FCB $00
5388 3A 04560 FCB $3A
5389 00 04570 FCB $00
538A 18 04580 FCB $18
538B 18 04590 FCB $18
538C 00 04600 FCB $00
538D 18 04610 FCB $18
538E 18 04620 FCB $18
538F 00 04630 FCB $00
5390 3F 04640 FCB $3F
5391 3C 04650 FCB $3C
5392 66 04660 FCB $66
5393 06 04670 FCB $06
5394 0C 04680 FCB $0C
5395 18 04690 FCB $18
5396 00 04700 FCB $00
5397 18 04710 FCB $18
5398 2D 04720 FCB $2D
5399 00 04730 FCB $00
539A 00 04740 FCB $00
539B 00 04750 FCB $00
539C 7E 04760 FCB $7E
539D 00 04770 FCB $00
539E 00 04780 FCB $00
539F 00 04790 FCB $00
53A0 2E 04800 FCB $2E
53A1 00 04810 FCB $00
53A2 00 04820 FCB $00
53A3 00 04830 FCB $00
53A4 00 04840 FCB $00
53A5 00 04850 FCB $00
53A6 60 04860 FCB $60
53A7 60 04870 FCB $60
53A8 2C 04880 FCB $2C
53A9 00 04890 FCB $00
53AA 00 04900 FCB $00
53AB 00 04910 FCB $00
53AC 00 04920 FCB $00
53AD 08 04930 FCB $08
53AE 18 04940 FCB $18
53AF 30 04950 FCB $30
53B0 61 04960 FCB $61
53B1 30 04970 FCB $30
53B2 48 04980 FCB $48
53B3 78 04990 FCB $78
53B4 48 05000 FCB $48
53B5 48 05010 FCB $48
53B6 62 05020 FCB $62
53B7 70 05030 FCB $70
53B8 48 05040 FCB $48
53B9 70 05050 FCB $70
53BA 48 05060 FCB $48
53BB 70 05070 FCB $70
53BC 63 05080 FCB $63
53BD 30 05090 FCB $30
53BE 48 05100 FCB $48
53BF 40 05110 FCB $40
53C0 48 05120 FCB $48
53C1 30 05130 FCB $30
53C2 64 05140 FCB $64
53C3 70 05150 FCB $70
53C4 28 05160 FCB $28
53C5 28 05170 FCB $28
53C6 28 05180 FCB $28
53C7 70 05190 FCB $70
53C8 65 05200 FCB $65
53C9 78 05210 FCB $78
53CA 40 05220 FCB $40
53CB 78 05230 FCB $78
53CC 40 05240 FCB $40
53CD 78 05250 FCB $78
53CE 66 05260 FCB $66
53CF 78 05270 FCB $78
53D0 40 05280 FCB $40
53D1 78 05290 FCB $78
53D2 40 05300 FCB $40
53D3 40 05310 FCB $40
```

```
53D4 67 05320 FCB $67
53D5 30 05330 FCB $30
53D6 48 05340 FCB $48
53D7 40 05350 FCB $40
53D8 58 05360 FCB $58
53D9 30 05370 FCB $30
53DA 68 05380 FCB $68
53DB 48 05390 FCB $48
53DC 48 05400 FCB $48
53DD 78 05410 FCB $78
53DE 48 05420 FCB $48
53DF 48 05430 FCB $48
53E0 69 05440 FCB $69
53E1 70 05450 FCB $70
53E2 20 05460 FCB $20
53E3 20 05470 FCB $20
53E4 20 05480 FCB $20
53E5 70 05490 FCB $70
53E6 6A 05500 FCB $6A
53E7 08 05510 FCB $08
53E8 08 05520 FCB $08
53E9 08 05530 FCB $08
53EA 48 05540 FCB $48
53EB 30 05550 FCB $30
53EC 6B 05560 FCB $6B
53ED 48 05570 FCB $48
53EE 50 05580 FCB $50
53EF 60 05590 FCB $60
53F0 50 05600 FCB $50
53F1 48 05610 FCB $48
53F2 6C 05620 FCB $6C
53F3 40 05630 FCB $40
53F4 40 05640 FCB $40
53F5 40 05650 FCB $40
53F6 40 05660 FCB $40
53F7 78 05670 FCB $78
53F8 6D 05680 FCB $6D
53F9 44 05690 FCB $44
53FA 6C 05700 FCB $6C
53FB 54 05710 FCB $54
53FC 44 05720 FCB $44
53FD 44 05730 FCB $44
53FE 6E 05740 FCB $6E
53FF 48 05750 FCB $48
5400 68 05760 FCB $68
5401 58 05770 FCB $58
5402 48 05780 FCB $48
5403 48 05790 FCB $48
5404 6F 05800 FCB $6F
5405 78 05810 FCB $78
5406 48 05820 FCB $48
5407 48 05830 FCB $48
5408 48 05840 FCB $48
5409 78 05850 FCB $78
540A 70 05860 FCB $70
540B 70 05870 FCB $70
540C 48 05880 FCB $48
540D 70 05890 FCB $70
540E 40 05900 FCB $40
540F 40 05910 FCB $40
5410 71 05920 FCB $71
5411 30 05930 FCB $30
5412 48 05940 FCB $48
5413 48 05950 FCB $48
5414 58 05960 FCB $58
5415 34 05970 FCB $34
5416 72 05980 FCB $72
5417 70 05990 FCB $70
5418 48 06000 FCB $48
5419 70 06010 FCB $70
541A 50 06020 FCB $50
541B 48 06030 FCB $48
541C 73 06040 FCB $73
541D 38 06050 FCB $38
541E 40 06060 FCB $40
541F 30 06070 FCB $30
5420 08 06080 FCB $08
5421 70 06090 FCB $70
5422 74 06100 FCB $74
5423 7C 06110 FCB $7C
5424 10 06120 FCB $10
5425 10 06130 FCB $10
5426 10 06140 FCB $10
5427 10 06150 FCB $10
5428 75 06160 FCB $75
5429 48 06170 FCB $48
542A 48 06180 FCB $48
542B 48 06190 FCB $48
542C 48 06200 FCB $48
542D 78 06210 FCB $78
542E 76 06220 FCB $76
542F 44 06230 FCB $44
5430 44 06240 FCB $44
5431 44 06250 FCB $44
5432 28 06260 FCB $28
5433 10 06270 FCB $10
5434 77 06280 FCB $77
5435 44 06290 FCB $44
5436 44 06300 FCB $44
5437 44 06310 FCB $44
5438 54 06320 FCB $54
5439 6C 06330 FCB $6C
543A 78 06340 FCB $78
543B 44 06350 FCB $44
543C 28 06360 FCB $28
543D 10 06370 FCB $10
543E 28 06380 FCB $28
543F 44 06390 FCB $44
5440 79 06400 FCB $79
5441 44 06410 FCB $44
5442 28 06420 FCB $28
5443 10 06430 FCB $10
5444 10 06440 FCB $10
5445 10 06450 FCB $10
5446 7A 06460 FCB $7A
5447 7C 06470 FCB $7C
5448 08 06480 FCB $08
5449 10 06490 FCB $10
544A 20 06500 FCB $20
544B 7C 06510 FCB $7C
544C 2F 06520 FCB $2F
544D 30 06530 FCB $30
544E 48 06540 FCB $48
544F 48 06550 FCB $48
5450 48 06560 FCB $48
5451 30 06570 FCB $30
5452 21 06580 FCB $21
5453 10 06590 FCB $10
5454 30 06600 FCB $30
5455 10 06610 FCB $10
5456 10 06620 FCB $10
5457 38 06630 FCB $38
5458 22 06640 FCB $22
5459 30 06650 FCB $30
545A 48 06660 FCB $48
545B 10 06670 FCB $10
545C 20 06680 FCB $20
545D 78 06690 FCB $78
545E 23 06700 FCB $23
545F 30 06710 FCB $30
5460 48 06720 FCB $48
5461 10 06730 FCB $10
5462 48 06740 FCB $48
5463 30 06750 FCB $30
5464 24 06760 FCB $24
5465 18 06770 FCB $18
5466 28 06780 FCB $28
5467 78 06790 FCB $78
5468 08 06800 FCB $08
5469 08 06810 FCB $08
546A 3D 06820 FCB $3D
546B 78 06830 FCB $78
546C 60 06840 FCB $60
546D 10 06850 FCB $10
546E 48 06860 FCB $48
546F 30 06870 FCB $30
5470 26 06880 FCB $26
5471 30 06890 FCB $30
5472 40 06900 FCB $40
5473 70 06910 FCB $70
5474 48 06920 FCB $48
5475 30 06930 FCB $30
5476 2B 06940 FCB $2B
5477 78 06950 FCB $78
5478 08 06960 FCB $08
5479 08 06970 FCB $08
547A 10 06980 FCB $10
547B 20 06990 FCB $20
547C 28 07000 FCB $28
547D 30 07010 FCB $30
547E 48 07020 FCB $48
547F 30 07030 FCB $30
5480 48 07040 FCB $48
5481 30 07050 FCB $30
5482 29 07060 FCB $29
5483 30 07070 FCB $30
5484 48 07080 FCB $48
5485 28 07090 FCB $28
5486 08 07100 FCB $08
5487 30 07110 FCB $30
```

```
5488      12      07120          FCB $12 END OF DATA
                  07126 * ROUTINE TO PREVENT THE ERASURE OF DESCENDERS *
5489 C6   05      07130          LDB #$05
548B E7   8D FD91 07140          STB $5220,PCR
548F 17   FD79    07150          LBSR $520B
5492 C6   07      07160          LDB #$07
5494 E7   8D FD88 07170          STB $5220,PCR
5498 C6   03      07180          LDB #$03
549A 39           07190          RTS
                  07195 * SCREEN REVERSAL ROUTINE *
549B 8E   0600    07200          LDX #$0600
549E A6   00      07210          LDA 0,X
54A0 43           07220          COMA
54A1 A7   80      07230          STA ,X+
54A3 8C   1E00    07240          CMPX #$1E00
54A6 26   F6      07250          BNE $549E
54A8 39           07260          RTS
                  07265 * U.S. MAP STORAGE ROUTINE *
54A9 108E 6000    07270          LDY #$6000
54AD 8E   13A6    07280          LDX #$13A6
54B0 5F           07290          CLRB
54B1 A6   85      07300          LDA B,X
54B3 5C           07310          INCB
54B4 A7   A0      07320          STA ,Y+
54B6 C1   14      07330          CMPB #$14
54B8 2D   F7      07340          BLT $54B1
54BA C6   20      07350          LDB #$20
54BC 3A           07360          ABX
54BD 8C   1C86    07370          CMPX #$1C86
54C0 2D   EE      07380          BLT $54B0
54C2 39           07390          RTS
                  07395 * U.S. MAP RESTORE ROUTINE *
54C3 108E 6000    07400          LDY #$6000
54C7 8E   13A6    07410          LDX #$13A6
54CA 5F           07420          CLRB
54CB A6   A0      07430          LDA ,Y+
54CD A7   85      07440          STA B,X
54CF 5C           07450          INCB
54D0 C1   14      07460          CMPB #$14
54D2 2D   F7      07470          BLT $54CB
54D4 C6   20      07480          LDB #$20
54D6 3A           07490          ABX
54D7 8C   1C86    07500          CMPX #$1C86
54DA 2D   EE      07510          BLT $54CA
54DC 39           07520          RTS
          0000    07530          END
00000 TOTAL ERRORS
```

50	0162	490	18C3
100	03F1	620	1B6D
140	06F6	740	1E99
190	0AC2	800	22EB
250	0EC8	880	24AE
290	11BE	980	26DD
340	1661	END	2A6D

Listing 2

```
10 '              WARGAME
20 '          WAR ROOM DRAWING
30 PCLEAR4:PMODE4,1:PCLS:POKE654
95,0:DEFUSR0=&H5120:DEFUSR1=&H51
41:CLS0:A=USR1(0):PRINT@70,"war"
;:PRINT@74,"room";:PRINT@79,"bei
ng";:PRINT@85,"drawn";:A=USR0(0)
:A=.708333333
40 READB,C:IFB=-1ANDC=-1THEN360E
LSEIFB=99ANDC=99THEND=0:E=0:GOTO
40
50 C=192-((C+SGN(C)*(ABS(C/8)+AB
S(C*C/150)))/1.1*A+96):B=B*15*A:
IFD=0ANDE=0THEND=B:E=C:GOTO40ELS
ELINE(D,E)-(B,C),PSET:D=B:E=C:GO
TO40
60 DATA0,25,.5,25,.75,21,.83,22,
1.15,8,1.4,10,1.4,15.5,1.9,22,2,
22,2.1,23,2.16,20,2.3,19
70 DATA2.3,15.2,2.5,15.8,2.6,11,
2.56,8.5,2.65,8.7,2.74,3,2.9,6,2
.7,7,2.65,11
80 DATA2.74,13,2.83,12.5,2.83,11
,3,10.5,3,8,3.3,11.5,3.32,14,3.1
5,18,3.1,20.5,3.23,22,3.38,22,3.
56,23,3.65,23.5,4,25.5,4,26.5,4.
13,29
90 DATA 4.11,31.5,3.95,36,4.2,37
.5,4.05,38,3.9,37.2,3.77,40,3.87
,40,4.1,41,4.1,39.5,4.35,40,4.35
,38,4.5,38,4.44,35,4.6,36,4.6,37
.5,4.54,38.5
100 DATA4.8,43,4.95,42.8,5.4,49,
5.47,52,5.47,53,5.4,54,5.15,53.5
,5.57,59.7,5.95,59.5,5.97,60,6.1
5,59,6.32,59.7,6.3,60,6.5,62,6.7
5,62.3,6.75,60.5,6.9,62,6.9,61,6
.45,56,6.5,51,6.6,52,6.67,53,6.8
,53,6.8,58,6.9,60,7.45,60,8,62.5
110 DATA7.72,65,7.9,64.6,8.05,65
.5,8.25,65.8,8.5,64.8,8.5,65.3,8
.65,65.3,8.5,66.7,8.4,66.6,8.38,
67.2,8.13,68,7.4,70,7.5,68.5,7.3
5,68,7.28,69.5,6.7,69.5,6.7,71,6
.2,71,6,71.5,5.35,72
120 DATA5.37,70,4.9,70,4.85,69,4
```

```
.6,71.5,3.5,73.5,3,72,3.5,75,3.4
8,76.5,3.15,76.5,3,77.5,2.73,77,
2.75,76.5,2,75.5,1.7,74.5,1.7,74
,1.47,74,1.47,73,.6,73.3,.5,71.5
,.5,69,0,70
130 DATA99,99,11,69,10.45,70,9.5
5,71,9.15,70,9.1,69,8.95,69,8.9,
68.5,9.1,67.5,9.2,67.5,9.2,66.5,
9.07,66.5,9.05,67,8.85,66.3,8.8,
66.2,8.9,65.5,9.2,65.7,9.3,66,9.
35,64.5,9.27,64.5,8.9,62,9,60.5,
9.2,60,9.25,60,9.3,60.5
140 DATA9.3,58.5,9.4,59,9.55,58.
5,9.5,57,9.2,55,9,54,9.48,56,9.7
5,59,9.75,60,9.95,61.5,9.8,60.5,
9.8,59.3,10.2,61,10.35,61,10.4,6
0,10.65,59.5,10.9,58.5,11,59,11.
1,59,11.45,51.5,11.8,50.5,11.85,
49.5,11.7,49.5,11.75,47
150 DATA11.7,42.5,11.77,38.5,11.
85,37.5,12,35,12.2,34,12.35,29,1
2.4,28.5,12.45,27,12.42,27,12.42
,27.5,12.5,25,12.55,24,12.6,22.5
,12.67,23,12.55,25.5,12.4,30,12.
38,32,12.48,32,12.51,30,12.75,26
,12.75,25,12.92,22,13,19,13.3,16
.5
160 DATA13.6,16,13.7,16.5,13.97,
14,14,14,14.15,13,14.25,11,14.25
,10.5,14.47,9,14.65,8.5,14.77,9,
14.77,8,14.8,7.5,14.82,5
170 DATA16.7,0,16.6,0,16.66,1.5,
16.62,1.5,16.6,5,16.55,5.5,16.2,
6,15.8,11,15.5,10.5,15.45,11.5,1
5.3,11.5,15.3,12.5,14.95,11.5,14
.8,9,14.77,10,14.5,10,14.45,11,1
4.47,15,14.45,16,14.1,16,14.2,17
,14.25,21.5,14,21.5,13.95,19.5
180 DATA13.7,19,13.6,20,13.5,23,
13.55,28,13.6,30,13.9,30,14,29,1
4.1,29,14,31,14.35,31,14.4,30,14
.45,30,14.47,27,14.55,25,14.6,25
,14.6,27.5,14.55,31,14.9,37.5,14
.9,38.5
190 DATA15.1,40,15.4,41,15.37,41
,15.34,42,15.5,44.5,15.6,46,15.5
3,44.5,15.45,44,15.8,46,15.65,46
,15.6,49,15.5,48.5,15.55,49,16,4
9,16.1,50,16.3,50.3,16.3,52,16.2
,52,16.2,53,16,53,15.7,60,15.6,5
9.7,15.6,59,15.45,59,15.45,61,15
.3,61.5,14.8,62,14.8,60
200 DATA14.75,59,14.8,58,14.85,5
6.5,14.75,55,14.8,51.5,14.7,51.5
,14.65,52.5,14.65,55,14.3,56,13.
95,57,13.85,57,13.85,59,13.75,59
,13.77,62,14,64.5,14.2,65.5,14.5
5,66,14.5,68.5,14.35,69,14.25,67
,14.2,68.5,14,67.5,13.8,71,13.6,
70,13.8,68,12.9,68.5,12.35,68
210 DATA12.4,69,11.5,70.5,11,69,
99,99
220 DATA2.45,5,3,-6,3.1,-3,2.6,5
,2.45,5,99,99,3.35,1,3.4,-2,3.8,
-2.5,3.95,.5,3.95,5.5,3.85,6,3.3
5,1,99,99,3.1,-6.5,99,99
230 DATA17.6,84,15.5,81,15.6,80,
15.1,79.5,15.5,76.5,16,77,16.55,
63,16.75,61,17.1,60.5,17.3,64.5,
18.3,68.5,18.7,80,19.2,81,17.6,8
4,99,99
240 DATA24,69,23.6,68.5,23.65,69
,23.25,68,23.25,67.5,23,68,23.1,
68.5,22.9,68.75,22.95,67,22.7,66
.5,22.75,65.5,22.5,66,22.55,65,2
2.15,67,22.6,66.5,22.7,67,21.9,7
1,21,69,20.75,64.5,20.4,63,20.4,
58,20.7,59,20.9,56,21.1,56.5,21.
15,59,21.25,59.5,21.2,60.5
250 DATA21.2,62.5,21.47,65,21.7,
65,21.72,64.5,21.45,63,21.55,60,
22,60,21.6,59,21.65,57.5,21.5,58
,21.4,55.5,20.7,55.5,20.75,57,20
.6,56.5,20.7,55,20.4,54,19.75,49
.5,19.9,46,19.87,43.5,19.4,44,19
.45,42.5,19.45,38,19.55,38.5,19.
6,37,19.85,37.5,20.2,42
260 DATA20.2,44,20.7,45,20.73,43
,21.1,40.5,21.03,39,21.15,40,21.
```

```
15,41,21.25,40.5,20.8,45,21,45,2
1.35,42,21.45,36,21.6,36.5,21.55
,39,21.8,40,21.85,36,22.45,36,22
.3,31,21.8,31,21.4,32.5,21.4,30,
20.8,32.5,20.75,36,19.85,35,19.6
,35.5,19.45,33,19.35,29
270 DATA19.15,28,18.8,21,18.9,21
,18.9,16,18.8,16,19,12,19.5,5,20
.25,6.5,20.4,5,20.6,5,20.65,4,20
.6,-1
280 DATA22.7,-1,23.2,4,23.4,12,2
3,11,22.7,15,22.2,30,22.25,28.5,
22.35,29,22.8,17,22.9,13,23.7,17
.5,24,23,23.7,25,23.6,24,23.25,3
0,23.4,30,23.7,26.5,23.8,27,23.8
2,26,24,26,99,99
290 DATA19.7,58,19.6,57,19.7,54,
19.77,54,19.6,51,20.05,52,20.1,5
3,19.8,55,19.9,57,19.75,57,19.8,
58,19.7,58,99,99,19.5,54.5,19.4,
54.5,19.4,51.5,19.55,52,19.6,54,
19.5,54.5,99,99,18.5,66,18.4,65.
5,18.7,63,19.1,64.5,19,65.5,18.5
5,65,18.5,66,99,99
300 DATA15.55,62,15.65,64,15.5,6
6,15.9,66,14.9,74,14.1,74,14,73,
14.55,71,14.97,71.5,15.05,70,15,
65,14.8,65,14.8,64.5,15.1,64.5,1
5.55,62,99,99,15.2,84,13.75,82,1
3.85,79,14.25,79,13.9,77,13.6,77
.3,13.9,76.5,13.9,75,14.6,75,14.
65,75.7,14,76,14.65,76.5
310 DATA15,79,15.9,83.5,15.2,84,
99,99,13.25,71,12,74.5,11.7,74.5
,11.75,73,12,73.5,12.4,70,13.25,
71,99,99,5.5,45,5.7,43.5,5.6,42,
5.45,42.5,5.5,45,99,99,5.4,41,5.
5,40,5.4,36,4.7,32,4.6,33,5.3,37
,5.4,41,99,99,5.55,54,5.6,49,5.5
5,49,5.5,46,5.5,52.5,5.55,54
320 DATA99,99,2.4,81,2.1,80.5,2.
7,78.5,2.65,78,3,78.5,3,79,2.9,7
9.5,2.85,79,2.85,80,2.4,81,99,99
,1.4,10.5,1.4,7.5,1.5,7,1.5,9.5,
1.4,10.5,99,99,9.7,19,9.65,19,9.
65,20,9.7,19,99,99,20.75,79,20.9
,78,21.2,78,20.9,77.5,21.15,77,2
1.3,78,21.4,78,21.4,77.5
330 DATA21.55,77.5,21.1,79,21.75
,79,21.85,80,21.2,80,20.75,79,99
,99,24,74.5,23.75,73,23.75,72,23
.8,71,23.5,71.5,23.75,75,24,76,9
9,99,0,76,.7,77,.7,76.5,0,75,99,
99,22,47,21.8,42,21.9,41,22.45,4
2,22.55,41,22.8,41.5,22.8,42.5,2
2.55,45,22.65,47,22.3,46
340 DATA22.5,45,22.25,45,22.1,46
.5,22,47,99,99,14.4,22.5,14.55,2
2.5,15.05,20,14.8,19.5,14.55,22,
14.4,22,99,99,15.1,18,15.4,18.5,
15.1,20,15.1,18,99,99,16.25,50,1
6.05,47.5,16.5,47,16.22,49,16.25
,50,99,99
350 DATA23.5,47.5,23.2,46.7,23.1
,45,23.4,40,23.3,37,23.55,35.5,2
3.65,36,23.6,40,23.7,39.5,23.65,
42,23.4,45,23.65,45.5,23.5,47.5,
-1,-1
360 GOTO680
370 LINE(0,93)-(255,93),PSET:LIN
E(0,94)-(255,94),PSET
380 LINE(0,95)-(255,95),PSET
390 LINE(0,109)-(255,109),PSET
400 LINE(207,110)-(207,179),PSET
410 LINE(207,179)-(255,179),PSET
420 LINE(207,119)-(255,119),PSET
430 LINE(0,179)-(255,179),PSET
440 PRINT@256,STRING$(32," ")
450 X1=125:Y1=52
460 READX,Y:IFX=99ANDY=99THEN480
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO460
470 DATA145,52,147,52,148,51,147
,53,148,56,149,56,150,52,152,53,
152,56,154,57,159,55,160,55,165,
51,99,99
480 X1=143:Y1=75
490 READX,Y:IFX=99ANDY=99THEN510
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
```

```
Y1=Y:GOTO490
500 DATA140,72,139,72,138,72,136
,70,133,70,99,99
510 LINE(114,26)-(114,37),PSET
520 X1=148:Y1=51
530 READX,Y:IFX=99ANDY=99THEN550
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO530
540 DATA149,50,150,50,151,51,155
,52,153,55,99,99
550 X1=225:Y1=43
560 READX,Y:IFX=99ANDY=99THEN580
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO560
570 DATA227,44,227,52,224,54,231
,54,99,99
580 LINE(235,24)-(233,27),PSET:L
INE(233,27)-(233,36),PSET:LINE(2
42,58)-(248,60),PSET:LINE(250,60
)-(255,60),PSET
590 X1=0:Y1=60
600 READX,Y:IFX=99ANDY=99THEN620
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO600
610 DATA2,62,4,60,9,60,8,58,11,5
7,12,56,12,54,15,54,16,52,19,52,
19,50,27,47,27,45,29,45,31,47,33
,48,35,47,36,48,37,47,40,47,42,5
0,44,50,44,51,52,49,51,51,52,51,
50,53,52,55,99,99
620 READX,Y:IFX=99ANDY=99THEN650
ELSEPSET(X,Y,1):GOTO620
630 DATA238,42,232,40,238,45,240
,46,238,47,242,31,48,34,70,42,53
,53,230,46
640 DATA99,27,107,29,128,62,132,
66,130,55,136,61,142,70,154,66,1
47,58,126,54,99,99
650 READX,Y:IFX=99ANDY=99THEN670
ELSEPAINT(X,Y),1,1:GOTO650
660 DATA0,63,28,89,58,56,139,75,
163,91,119,26,131,21,146,24,152,
28,154,20,154,12,171,12,199,31,2
23,31,223,48,221,56,235,56,228,5
6,228,62,235,61,209,68,99,99
670 PRINT@480,"";:GOTO730
680 '
690 PRINT@314,"DEFCON";:PRINT@35
0,"5";:PRINT@382,"4";:PRINT@414,
"3";:PRINT@446,"2";:PRINT@478,"1
";:PRINT@480,"";
700 LINE(210,121)-(253,121),PSET
:LINE(253,121)-(253,131),PSET:LI
NE(253,131)-(210,131),PSET:LINE(
210,131)-(210,121),PSET:LINE(234
,131)-(234,121),PSET:PAINT(211,1
22),1,1
710 PRINT@288,"LAUNCH";:PRINT@32
1,"CODE";:LINE(48,109)-(48,179),
PSET
```

```
720 GOTO370
730 PRINT@393,"norad":PRINT@329,
CHR$(34)"&":PRINT@332,CHR$(34)"=
":PRINT@368,CHR$(34)"$":PRINT@40
3,CHR$(34)"/":X1=68:Y1=112
740 READX,Y:IFX=99ANDY=99THEN810
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO740
750 DATA128,112,136,114,132,116,
132,118,135,117,135,119,140,114,
140,116,145,116,142,118,141,121,
145,119,141,121,142,123,144,124,
148,123,148,119,150,117,152,118,
153,118,154,119,154,121,152,122,
155,122,152,124
760 DATA154,125,162,124,162,122,
165,122,166,123,170,120,170,119,
174,117,178,117,180,112,181,114,
183,111,185,114,187,113,183,118,
184,120,181,123,181,125,178,127,
178,129,175,131,175,133,177,135,
178,137,175,140
770 DATA173,141,171,142,171,144,
170,146,169,148,168,152,169,156,
170,158,173,161,176,165,175,167,
172,165,169,163,167,162,167,161,
165,159,163,158,161,158,159,160,
157,158,147,158,148,160,149,158,
147,161,146,161,138,159
```



```
780 DATA135,159,130,160,128,161,
126,162,126,164,125,166,126,168,
124,168,122,166,120,165,119,163,
117,161,114,161,112,163,110,162,
108,160,104,159,100,159,100,160,
92,160,80,157,75,157,73,154,70,1
53,70,151,67,150
790 DATA65,145,64,140,62,139,64,
138,62,135,63,132,64,126,65,120,
63,115,62,113,67,118,68,118,69,1
17,69,115,68,112
800 DATA67,109,66,111,64,111,62,
109,140,109,136,114,140,109,142,
110,143,112,144,114,146,114,149,
115,153,115,155,116,156,118,156,
120,157,122,160,122,160,120,169,
120,99,99
810 LINE(187,113)-(190,109),PSET
:X1=126:Y1=168
820 READX,Y:IFX=99ANDY=99THEN840
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO820
830 DATA128,169,128,170,126,171,
129,179,99,99
840 X1=75:Y1=157
850 READX,Y:IFX=99ANDY=99THEN870
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO850
860 DATA76,163,77,165,80,170,80,
172,78,172,80,174,82,175,83,179,
86,179,85,175,86,173,84,170,83,1
65,82,162,82,161,86,162,89,163,8
9,165,90,170,92,175,94,179,99,99
870 X1=150:Y1=179
880 READX,Y:IFX=99ANDY=99THEN900
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO880
890 DATA152,177,154,176,156,176,
157,177,158,176,159,177,158,178,
99,99
900 X1=166:Y1=174
910 READX,Y:IFX=99ANDY=99THEN930
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO910
920 DATA168,173,171,172,175,172,
176,173,179,174,183,175,185,175,
184,176,187,176,189,177,188,178,
178,178,180,177,176,176,173,175,
174,174,172,174,165,175,99,99
930 READX,Y:IFX=99ANDY=99THEN950
ELSEPSET(X,Y,1):GOTO930
940 DATA67,140,77,155,76,118,105
,135,125,159,167,145,141,125,72,
117,99,99
950 X1=64:Y1=131
960 READX,Y:IFX=99ANDY=99THEN980
ELSELINE(X1,Y1)-(X,Y),PSET:X1=X:
Y1=Y:GOTO960
970 DATA67,131,70,133,122,133,12
2,154,120,154,120,163,99,99
```

```
980 X1=122:Y1=154
990 READX,Y:IFX=99ANDY=99THEN101
0ELSELINE(X1,Y1)-(X,Y),PSET:X1=X
:Y1=Y:GOTO990
1000 DATA147,154,147,139,156,139
,162,133,169,117,169,110,99,99
1010 LINE(169,117)-(183,117),PSE
T
1020 LINE(162,133)-(174,133),PSE
T
1030 LINE(122,110)-(122,133),PSE
T
1040 LINE(85,110)-(85,133),PSET
1050 READX,Y:IFX=99ANDY=99THEN10
70ELSELINE(X,Y)-(X,Y+3),PSET:LIN
E(X+2,Y)-(X+2,Y+3),PSET:LINE(X+4
,Y)-(X+4,Y+3),PSET:GOTO1050
1060 DATA109,132,115,135,92,156,
88,116,100,114,112,114,116,118,1
09,118,111,122,115,126,127,143,1
29,139,136,149,99,99
1070 READX,Y:IFX=99ANDY=99THEN10
90ELSELINE(X,Y)-(X+2,Y-2),PSET:L
INE(X+2,Y-2)-(X+4,Y),PSET:LINE(X
+4,Y)-(X,Y),PSET:GOTO1070
1080 DATA109,129,73,115,78,116,7
0,140,74,139,73,136,78,153,113,1
56,114,150,120,145,119,133,118,1
16,128,157,141,146,145,128,161,1
53,173,120,177,122,99,99
1090 LINE(105,138)-(109,138),PSE
T:LINE-(106,140),PSET:LINE-(107,
137),PSET:LINE-(108,140),PSET
1100 POKE65494,0:A=USR1(0):CLS0:
PRINT@256,"  INSERT TAPE TO SAVE
 WAR ROOM      PICTURE TO, THEN H
IT ENTER":PRINT"
                    ";:PRINT@334,"";
:INPUTA$:CSAVEM"WAR ROOM",1536,7
679,20768
```

```
150.....0243
230.....04B1
440.....074C
560.....0973
END...0B3C
```

Listing 3

```
10 '                WARGAME
20 '              INSTRUCTIONS
30 '
40 '              BY TOM WEBER
50 DEFUSR0=&H5120:DEFUSR1=&H5141
:CLS:POKE65495,0
60 INPUT"DO YOU WANT INSTRUCTION
S";A$:IFLEFT$(A$,1)="N"THEN220
70 M=1
80 CLS
90 ONM GOSUB140,150,160,170,180,
190,200,210
100 IFM=6THENM=M+1:GOTO90
110 FORA=1TOLEN(A$):B$=MID$(A$,A
,1):PRINTB$;:POKE140,190:POKE142
,2:EXEC43350:NEXT:M=M+1:PRINT:FO
RX=1TO300:NEXT
120 IFM<9THEN90
130 FORX=1TO300:NEXT:PRINTSTRING
$(32,"-");:FORX=1TO16:PRINT:FORY
=1TO100:NEXTY,X:GOTO220
140 PRINT@13,"";:A$="WARGAME":RE
TURN
150 A$="THIS PROGRAM IS A SIMULA
TION OF OPERATING OUR NORTH AMER
ICAN DEFENSE (NORAD) COMPLEX IN
CHEYENNE MOUNTAIN, COLORADO.":RE
TURN
160 A$="IN ACTUALITY, THIS IS A
SIMULATION OF A SIMULATION. ONE
OF A NUCLEAR ATTACK CONDUCTED BY
 NORAD TO TEST PERSONEL AND EQUI
PTMENT IN THE EVENT OF AN ACTUAL
 NUCLEAR WAR.":RETURN
170 A$="THE OBJECT OF THE SIMULA
TION IS SIMPLE: DISABLE ALL SOVI
ET BASES BEFORE THEY DISABLE OUR
S.":RETURN
180 PRINT"  ";:A$="GOOD LUCK, GE
NERAL. THE PRESIDENT EXPECTS A H
IGH RATING OF NORAD AS A RESULT
OF THIS TEST.":RETURN
190 FORX=1TO5:PRINT:FORY=1TO200:
NEXTY,X:RETURN
200 PRINT@498,"";:A$="THAT IS AL
```

```
L.":RETURN
210 PRINT@500,"";:A$="PENTAGON O
UT":RETURN
220 CLS0:PRINT@231,"SIMULATION B
EGINS";
230 POKE65494,0:FORPM=0TO4STEP2:
FORSC=0TO1:SCREEN0,SC:PLAY"V1503
L17EV31L2A":PMODEPM,1:SCREEN1,SC
:PLAY"O2V10L90EL15C":FORX=1TO520
:NEXTX,SC,PM
240 A=USR0(0):SCREEN1,1
250 POKE65495,0
260 A$="SIMULATION WILL BEGIN SH
ORTLY..... ":GOSUB270:GOTO340
270 L=LEN(A$):A=0:B=0
280 A=A+1
290 D$=MID$(A$,A,1)
300 L$="
       "+D$
310 PRINT@256,L$;:A=A+1:IFA>L TH
ENA=L:B=B+1:IFB=32THENRETURN
320 L$=RIGHT$(L$,LEN(L$)-1)+MID$
(A$,A,1)
330 SOUND200,2:GOTO310
340 'SHOW BASES
350 A$="SOVIET BASES:          ":X=0
:Y=0:GOSUB590
360 A$="MOSCOW   ":X=230:Y=42:GOS
UB590
370 A$="LENINGRAD   ":X=232:Y=40:
GOSUB590
380 A$="KHARKOV   ":X=238:Y=45:GO
SUB590
390 A$="VOLOGRAD   ":X=240:Y=46:G
OSUB590
400 A$="ROSTOV   ":X=238:Y=47:GOS
UB590
410 A$="ARCHANGEL   ":X=242:Y=31:
GOSUB590
420 A$="MINSK   ":X=230:Y=46:GOSU
B590
430 A$="KAMCHATKA   ":X=70:Y=42:G
OSUB590
440 A$="VLADIVOSTOK   ":X=53:Y=53
:GOSUB590
450 A$="YAKUTSK   ":X=48:Y=34:GOS
UB590
460 A$="U.S. BASES:          ":X=0:Y=
0:GOSUB590
470 A$="NOME   ":X=99:Y=27:GOSUB5
90
480 A$="FAIRBANKS   ":X=107:Y=29:
GOSUB590
490 A$="SEATTLE   ":X=126:Y=54:X2
=72:Y2=117:GOSUB590
500 A$="SPOKANE   ":X=130:Y=55:X2
=76:Y2=118:GOSUB590
510 A$="SAN FRANCISCO   ":X=128:Y
=62:X2=67:Y2=140:GOSUB590
520 A$="SAN DIEGO   ":X=132:Y=66:
X2=77:Y2=155:GOSUB590
530 A$="DENVER   ":X=136:Y=61:X2=
105:Y2=135:GOSUB590
540 A$="HOUSTON   ":X=142:Y=70:X2
=125:Y2=159:GOSUB590
550 A$="CHICAGO   ":X=147:Y=50:X2
=141:Y2=125:GOSUB590
560 A$="CHARLESTON   ":X=154:Y=66
:X2=167:Y2=145:GOSUB590
570 PRINT@276,"                ";
580 GOTO710
590 L=LEN(A$):A=0:B=0
600 A=A+1
610 D$=MID$(A$,A,1)
620 L$="
       "+D$
630 PRINT@256,L$;
640 IFY=0ORF1=1THEN660ELSEIFPPOI
NT(X,Y)=0THENPSET(X,Y,1)ELSEPRES
ET(X,Y)
650 IFY2=0THEN660ELSEIFPPOINT(X2
,Y2)=0THENPSET(X2,Y2,1)ELSEPRESE
T(X2,Y2)
660 A=A+1:IFA>L THENA=L:IFY=0ORF
1=1THEN670ELSEPSET(X,Y,1)ELSE680
670 PRINT@256,"
                  ":IFF1=1THENF1=0:
RETURNELSEIFY2=0THENRETURNELSEPS
ET(X2,Y2,1):RETURN
680 L$=RIGHT$(L$,LEN(L$)-1)+MID$
(A$,A,1)
690 SOUND200,2:GOTO630
700 PRINT@480,"";
710 PRINT@480,"";:POKE65494,0:CL
OAD
```

110....	027C	1340....	1707
225....	05B2	1420....	19B5
380....	078E	1460....	1BB2
570.....	0961	1520....	1E85
735....	0C64	1670....	2101
915....	0F53	1750....	236A
1040....	1162	1920....	262D
1240....	1490	END....	2833

Listing 4

```
10 '                WARGAME
20 '
30 '              BY TOM WEBER
40 POKE65495,0:GOSUB2020:GOTO90
50 L=LEN(A$):A=0:B=0:A=A+1:D$=MI
D$(A$,A,1):L$=STRING$(31," ")+D$
60 PRINT@256,L$;:A=A+1:IFA>L THE
NA=L:IFY=0ORF1=1THEN70ELSEPSET(X
,Y,1)ELSE80
70 PRINT@256,STRING$(32," "):RET
URN
80 L$=RIGHT$(L$,LEN(L$)-1)+MID$(
A$,A,1):SOUND200,2:GOTO60
90 SB$(1)="213778":SB$(2)="22378
4":SB$(3)="233790":SB$(4)="24379
C":SB$(5)="3D37A8":SB=RND(5):SS=
SB:RM=RND(2000)
```

```
100 UC=10000+RND(89999):CU=10000
+RND(89999):PRINT@383,UC;:PRINT@
415,CU;:CU$=STR$(CU)+STR$(UC):CU
$=RIGHT$(CU$,11):CU$=LEFT$(CU$,5
)+RIGHT$(CU$,5):FORSZ=1TO10:OC$(
SZ)=MID$(CU$,SZ,1):OC(SZ)=10:NEX
T:SU=10:US=10:TIMER=0:DIMX1(50),
Y1(50),X2(50),Y2(50)
101 POKE65495,0:GOTO340
110 'COMPUTER ARC DRAW
120 GOSUB1370:FORX=0TO50:X1(X)=0
:Y1(X)=0:NEXT:R=0:ZM=0:F=0:B=USR
0(0):PMODE4,1:SCREEN1,1:XX=X1:XO
=X1:YO=Y1:IFX2<X1 THENR=1:X=X1:Y
=Y1:X1=X2:Y1=Y2:X2=X:Y2=Y:XO=X2:
YO=Y2
130 X1(0)=X1:Y1(0)=Y1:XF=(X1+X2)
/2:YF=(Y1+Y2)/2:D=(Y1-Y2):AOS=XF
:XV=AOS:YV=(YF+D)/2
140 GOSUB990:IFZM>50THENZM=0
141 IFZM<0THENZM=50
145 XM=(X1+XV)/2:YM=(Y1+YV)/2:XN
=(X1+XM)/2:YN=YM:X1(ZM)=X1:Y1(ZM
)=Y1:IFF=0THENZM=ZM+1ELSEZM=ZM-1
150 IFINT(XN)=INT(XV)-2THEN180
160 IFINT(XN)=INT(XV)+3 THEN190
170 X1=XN:Y1=YN:GOTO140
180 ZM=ZM-1:IFZM<0THENZM=0
181 IFZM>47THENZM=47
185 F=1:X1(ZM)=XV:Y1(ZM)=YV:X1(Z
M+1)=XV:Y1(ZM+1)=YV:X1(ZM+2)=XV:
Y1(ZM+2)=YV:X1(ZM+3)=XV:Y1(ZM+3)
=YV:ZM=ZM+2:X1=X2:Y1=Y2:GOTO140
190 F=0:ZM=ZM+2:IFR=1THEN260
200 IFZM>50THENZM=50
205 IFY1(ZM)=0THENZM=ZM-1
210 F2=1:A=-1
220 A=A+1:IFA>ZM THEN240ELSEIFX1
(A)<0THENX1(A)=XO
221 IFY1(A)<0THENY1(A)=YO
225 PSET(X1(A),Y1(A),1):LINE(XO,
YO)-(X1(A),Y1(A)),PSET:XO=X1(A):
YO=Y1(A):IFUU=1THEN960ELSEGOSUB9
90:GOTO220
230 GOTO220
240 F=0:ZM=0:F2=0:RR=0:IFYO>52TH
EN670ELSEIFUS<1THEN1940
250 RETURN
260 '
270 IFY1(ZM)=0THENZM=ZM-1:GOTO27
0
280 A2=ZM+1
290 A2=A2-1:IFA2<0THEN310ELSEPSE
T(X1(A2),Y1(A2),1):LINE(XO,YO)-(
X1(A2),Y1(A2)),PSET:XO=X1(A2):YO
=Y1(A2):IFUU=1THEN960ELSEGOSUB99
0
300 GOTO290
```

```
310 F=0:ZM=0:RR=0:IFYO>52THEN670
ELSEIFUS<1THEN1940
320 RETURN
330 'GET SOVIET COORDINATES
340 RR=1:B=RND(10)
350 ONB GOSUB470,480,490,500,510
,520,530,540,550,560:IFU=1THENIF
SU(B)=0THENSU=SU-1
360 IFU=1THENSU(B)=1:X4=X7:Y4=Y7
:RETURN
370 X1=X7:Y1=Y7:TX=X1:TY=Y1
380 'GET U.S. COORDINATES
390 B=RND(10)
400 ONB GOSUB570,580,590,600,610
,620,630,640,650,660:IFU9=1THENI
FUS(B)=1THENGOSUB1870
410 IFU=1THENX5=X8:Y5=Y8:X6=X9:Y
6=Y9:RETURN
420 X2=X8:Y2=Y8:X3=X9:Y3=Y9:XT=X
2:YT=Y2:UX=X3:UY=Y3:IFUS(B)=0THE
NUS=US-1
430 US(B)=1:GOSUB110
440 PR=SU+1
450 PU=RND(PR):IFPU<>1 THENGOSUB
990:GOTO450
460 GOTO340
470 X7=238:Y7=42:RETURN
480 X7=232:Y7=40:RETURN
490 X7=238:Y7=45:RETURN
500 X7=240:Y7=46:RETURN
510 X7=238:Y7=47:RETURN
520 X7=242:Y7=31:RETURN
530 X7=48:Y7=34:RETURN
540 X7=70:Y7=42:RETURN
550 X7=53:Y7=53:RETURN
560 X7=230:Y7=46:RETURN
570 X8=99:Y8=27:RETURN
580 X8=107:Y8=29:RETURN
590 X8=128:Y8=62:X9=67:Y9=140:RE
TURN
600 X8=132:Y8=66:X9=77:Y9=155:RE
TURN
610 X8=130:Y8=55:X9=76:Y9=118:RE
TURN
620 X8=136:Y8=61:X9=105:Y9=135:R
ETURN
630 X8=142:Y8=70:X9=125:Y9=159:R
ETURN
640 X8=154:Y8=66:X9=167:Y9=145:R
ETURN
650 X8=147:Y8=58:X9=141:Y9=125:R
ETURN
660 X8=126:Y8=54:X9=72:Y9=117:RE
TURN
670 'U.S. GROUND EXPLOSION
680 IFU=1THENRETURN
690 LINE((XX+X3)/2,109)-(X3,Y3),
PSET:FORX=1TO4:CIRCLE(X3,Y3),X:N
EXT:POKE65494,0:PLAY"V60L9C":PO
KE65495,0:IFUS<1THEN1940ELSEGOTO
440
700 'PLAYER'S ARC DRAW
710 FORXL=0TO50:X2(XL)=0:Y2(XL)=
0:NEXT:R2=0:ZZ=0:QQ=0:XA=X4:XI=X
4:YI=Y4:IFX5<X4 THENR2=1:X=X4:Y=
Y4:X4=X5:Y4=Y5:X5=X:Y5=Y:XI=X5:Y
I=Y5
720 X2(0)=X4:Y2(0)=Y4:XF=(X4+X5)
/2:YF=(Y4+Y5)/2:D=(Y4-Y5):AOS=XF
:XV=AOS:YV=(YF+D)/2
730 IFZZ>50THENZZ=0
731 IFZZ<0THENZZ=50
735 XM=(X4+XV)/2:YM=(Y4+YV)/2:XN
=(X4+XM)/2:YN=YM:X2(ZZ)=X4:Y2(ZZ
)=Y4:IFQQ=0THENZZ=ZZ+1ELSEZZ=ZZ-
1
740 IFINT(XN)=INT(XV)-2THEN770
750 IFINT(XN)=INT(XV)+3 THEN790
760 X4=XN:Y4=YN:GOTO730
770 ZZ=ZZ-1:QQ=1:X2(ZZ)=XV:Y2(ZZ
)=YV:X2(ZZ+1)=XV:Y2(ZZ+1)=YV:X2(
ZZ+2)=XV:Y2(ZZ+2)=YV:X2(ZZ+3)=XV
:Y2(ZZ+3)=YV:ZZ=ZZ*2:X4=X5:Y4=Y5
:GOTO730
780 '
790 ZZ=ZZ*2:UU=1:IFR2=1THEN880
800 IFY2(ZZ)=0THENZZ=ZZ-1:GOTO80
0
810 IFU=1THENIFYU>52THENLINE(X6,
```

```
Y6)-((X2(0)+X6)/2,109),PSET
820 AU=-1:FK=1
830 AU=AU+1:IFAU>ZZ THEN850ELSEI
FX2(AU)<0THENX2(AU)=XI
835 PSET(X2(AU),Y2(AU),1):LINE(X
I,YI)-(X2(AU),Y2(AU)),PSET:XI=X2
(AU):YI=Y2(AU):GOSUB990:IFRR=1TH
EN970ELSEGOTO830
840 GOTO830
850 IFSU=0THEN1900
860 UU=0:FK=0:IFRR=1THEN970ELSER
ETURN
870 '
880 IFY2(ZZ)=0THENZZ=ZZ-1:GOTO88
0
890 IFU=1THENIFYU>52THENLINE(X6,
Y6)-((X2(0)+X6)/2,109),PSET
900 AU=ZZ+1
910 AU=AU-1:IFAU=-1THEN930ELSEIF
X2(AU)<0THENX2(AU)=XI
915 PSET(X2(AU),Y2(AU),1):LINE(X
I,YI)-(X2(AU),Y2(AU)),PSET:XI=X2
(AU):YI=Y2(AU):GOSUB990:IFRR=1TH
EN970ELSEGOTO910
920 GOTO910
930 IFSU=0THEN1900
940 UU=0:IFRR=1THEN970ELSERETURN
950 'RETURN TO OTHER'S ARC
960 IFFK=1THEN830ELSE910
970 IFF2=1THEN220ELSE290
980 '
990 'PLAYER INPUT
1000 IFLA=1THENV=RND(5):IFLL=9TH
ENV=1:V=RND(10):OC(V)=VAL(OC$(V)
):GOTO1020ELSEIFV=1THENV=RND(10)
:OC(V)=VAL(OC$(V)):GOTO1020ELSE1
020
1010 GOTO1150
1020 LL=0:FORSZ=10TO1STEP-1:IFOC
(SZ)<10THENV=OC(SZ):LL=LL+1ELSEV
=RND(9)
1030 V$=STR$(V):ONSZ GOSUB1100,1
110,1120,1130,1140,1050,1060,107
0,1080,1090:NEXT:IFLL=10THENLA=2
:GOTO1040ELSEGOTO1170
1040 A$="U.S. LAUNCH CODE RE-ACT
IVATED":IFU$=""THENGOSUB50:GOTO1
150ELSEFORSZ=1TOLEN(U$):PRINT@47
9+SZ,MID$(U$,SZ,1);:SOUND240,1:N
EXT:GOSUB50:GOTO1150
1050 PRINT@383,V$;:RETURN
1060 PRINT@384,V$;:RETURN
1070 PRINT@385,V$;:RETURN
1080 PRINT@386,V$;:RETURN
1090 PRINT@387,V$;:RETURN
1100 PRINT@415,V$;:RETURN
1110 PRINT@416,V$;:RETURN
1120 PRINT@417,V$;:RETURN
1130 PRINT@418,V$;:RETURN
1140 PRINT@419,V$;:RETURN
```

```
1150 IFLA=0ANDTIMER>1000+RM THEN
PRINT@383,10000+RND(89999):PRINT
@415,10000+RND(89999):LA=1:A$="S
OVIETS HAVE JAMMED YOUR LAUNCH C
ODE...":GOSUB50:PRINT@480,"** CH
ANGES LOCKED OUT **":SOUND5,2
1160 IFLA=1THENA$="COMPUTER ATTE
MPTING TO RESTORE CODE...STAND B
Y...":GOSUB50:PRINT@480,STRING$(
30," ")
1170 GOSUB1520:IFU=1THENRETURN
1180 IFSL=1THEN1610
1190 I$=INKEY$:IFI$=""THENRETURN
1200 SOUND240,1:IFI$>"@"ANDI$<CH
R$(91)ORI$>"/"ANDI$<":"ORI$="?"O
RI$=CHR$(13)ORI$=CHR$(8)ORI$="-"
THEN1210ELSERETURN
1210 IFI$=CHR$(8)THENIFI>0THENI=
I-1:U$=LEFT$(U$,I):PRINT@480+I,"
 ";:RETURN
1220 IFI$="?"ANDI=0THEN1310
1230 IFHP=1THENGOSUB1310
1240 IFPEEK(338)=191THENIFLA=1TH
ENU$="":I=0:PRINT@269,"STAND BY.
..":FORSZ=1TO4:PRINT@480,"** ACC
ESS DENIED **":PLAY"V2801L6C":PR
INT@480,STRING$(20," "):FORZS=1T
O150:NEXTZS,SZ:PRINT@269,"
     ":PRINT@480,STRING$(20," ")
:RETURNELSEGOTO1270
1250 U$=U$+I$:PRINT@480+I,I$;:I=
I+1:RETURN
1260 PRINT@480,STRING$(32," "):I
=0:U$="":RETURN
1270 U=INSTR(U$,CHR$(8)):IFU=0TH
EN1280ELSEU$=LEFT$(U$,U-1)+MID$(
U$,U+1):GOTO1270
1280 IFU$="JAM"THEN1350ELSEIFU$=
"SUBS?"THEN1390ELSEIFU$="USCON"T
HEN1430ELSEIFU$="SOVCON"THEN1470
1290 IFU$="TIME?"THEN1510ELSEIFL
EFT$(U$,6)="LAUNCH"THENU$=MID$(U
$,7):GOTO1720ELSEIFU$="DESUB"THE
N1590ELSEGOSUB1260:PRINT@480,"*C
OMMAND INPUT ERROR*":SOUND240,1:
SOUND50,1:GOSUB1260:RETURN
1300 'HELP MENU
1310 GOSUB1260:IFHP=1THEN1330
1320 PRINT@288,"USCON ":PRINT"SO
VCON":PRINT"JAM    ":PRINT"SUBS?
":PRINT"DESUB ":PRINT"LAUNCH":PR
INT"TIME? ":HP=1:RETURN
1330 HP=0:PRINT@288,"LAUNCH":PRI
NT" CODE ":PRINT"       ":PRINT"
     ":PRINT"       ":PRINT"
 ":PRINT@383,UC;:PRINT@415,CU;:L
INE(0,109)-(48,109),PSET:LINE(0,
179)-(48,179),PSET:GOSUB1260:RET
```

```
URN
1340 'JAM SOVIET'S CODE
1350 GOSUB1260:IFJM>0THENA$="SOV
IETS HAVE SECURED THEIR LAUNCH C
ODE FROM OUR SENSORS...JAMMING I
S INEFFECTIVE":GOSUB50:RETURN
1360 JM=1:A$="SOVIET LAUNCH CODE
 JAMMED":GOSUB50:J1=0:RETURN
1370 IFJM=1THENIFJ1<350THENJ1=J1
+1:GOSUB990:GOTO1370ELSEJM=2:A$=
"SOVIET LAUNCH CODE RE-ACTIVATED
":GOSUB50:RETURNELSERETURN
1380 '# OF SUBS
1390 GOSUB1260:A$="THERE ARE"+ST
R$(SS)+" ENEMY SUBS REMAINING":G
OSUB50:IFSS=0THENRETURN
1400 FORSZ=1TOSB:IFSB$(SZ)=""THE
NNEXT:RETURNELSESC=VAL("&H"+LEFT
$(SB$(SZ),2)):SX=VAL("&H"+MID$(S
B$(SZ),3,2)):SY=VAL("&H"+RIGHT$(
SB$(SZ),2)):PRINT@263+(SZ+1)*32,
CHR$(SC):LINE(SX,SY)-(SX+7,SY),P
SET:LINE-(SX+8,SY-1),PSET:LINE-(
SX+7,SY-2),PSET
1410 LINE-(SX+5,SY-2),PSET:LINE-
(SX+5,SY-3),PSET:LINE-(SX+4,SY-3
),PSET:LINE-(SX+4,SY-2),PSET:LIN
E-(SX,SY-2),PSET:LINE-(SX-1,SY-1
),PSET:LINE-(SX-2,SY-2),PSET:LIN
E-(SX-2,SY),PSET:SOUND200,1:NEXT
:RETURN
1420 'U.S. CONDITION
1430 GOSUB1260:A$="UNITED STATES
' CONDITION: ":GOSUB50:EXEC&H54A
9:FORSZ=294TO454STEP32:PRINT@SZ,
STRING$(20," "):NEXT:UH=RND(100)
:UP=RND(100000000):PRINT@294,"PR
OJECTED KILL RATIO":PRINT@333,UH
:PRINT@358,"% HOUSING DESTROYED"
:PRINT@394,"";:PRINTUSING"####
1440 FORSZ=1TO3000:NEXT:FORSZ=29
4TO454STEP32:PRINT@SZ,STRING$(20
," "):NEXT:PRINT@294,"BASES REMA
INING:":X=326:FORY=1TO10:IFUS(Y)
=1THENNEXTELSEPRINT@X,Y:SOUND10,
1:X=X+32:IFX=486THENX=335:NEXTEL
SENEXT
1450 FORSZ=1TO3000:NEXT:FORSZ=29
4TO454STEP32:PRINT@SZ,STRING$(20
," "):NEXT:PRINT@294,"MISSLES TA
RGETED":PRINT@330,"AND READY":FO
RSZ=1TO1000:NEXT:EXEC&H54C3:RETU
RN
1460 'SOVIET CONDITION
1470 GOSUB1260:A$="SOVIET UNION'
S CONDITION: ":GOSUB50:EXEC&H54A
9:FORSZ=294TO454STEP32:PRINT@SZ,
STRING$(20," "):NEXT:SH=RND(100)
```

```
:SP=RND(100000000):PRINT@294,"PR
OJECTED KILL RATIO":PRINT@333,SH
:PRINT@358,"% HOUSING DESTROYED"
:PRINT@394,"";:PRINTUSING"####
1480 FORSZ=1TO3000:NEXT:FORSZ=29
4TO454STEP32:PRINT@SZ,STRING$(20
," "):NEXT:PRINT@294,"BASES REMA
INING:":X=326:FORY=1TO10:IFSU(Y)
=1THENNEXTELSEPRINT@X,Y:SOUND100
,1:X=X+32:IFX=486THENX=335:NEXTE
LSENEXT
1490 FORSZ=1TO3000:NEXT:FORSZ=29
4TO454STEP32:PRINT@SZ,STRING$(20
," "):NEXT:PRINT@294,"MISSLES TA
RGETED":PRINT@330,"AND READY":FO
RSZ=1TO1000:NEXT:EXEC&H54C3:RETU
RN
1500 'TIME FUNCTION
1510 GOSUB1260:GOSUB1560:FORSZ=2
94TO454STEP32:PRINT@SZ,STRING$(2
0," "):NEXT:PRINT@298,"GAME TIME
":PRINT@330," ELAPSED":PRINT@358
,THE"HRS"TME"MIN"TSE"SEC":PRINT@
394,"ESTIMATED TIME":PRINT@428,"
REMAINING":PRINT@454,RH"HRS"RM"M
IN"RS"SEC":FORSZ=1TO4000:NEXT:
1520 '
1530 IFTIMER>32768THENTF=1
1540 IFTIMER<32768 AND TF=1THENF
4=F4+1:TF=0
1550 GOSUB1780:RETURN
1560 EXEC&H54A9:TS=((F4*18)*60)+
FIX(TIMER/60.680555):THE=FIX(TS/
60/60):TS=FIX(TS-(THE*60*60)):TM
E=FIX(TS/60):TS=FIX(TS-TME*60):T
SE=TS:RH=4-THE:RM=59-TME:RS=60-T
SE:RETURN
1570 '
1580 'SUB DESTROY MODE
1590 GOSUB1260:A$="SUBMARINE DES
TROY MODE":GOSUB50
1600 XS=128:YS=144:SL=1:S1=XS:S2
=YS:C=PPOINT(XS,YS)
1610 PSET(S1,S2,C):C=PPOINT(XS,Y
S):S1=XS:S2=YS:IFC=0THENPSET(XS,
YS,1)ELSEPRESET(XS,YS)
1620 GOSUB1680:I$=INKEY$:IFPEEK(
338)=191THEN1700ELSEIFPEEK(340)=
191THENSL=0:PSET(XS,YS,C):RETURN
1630 IFPEEK(341)=247THENYS=YS-1:
IFYS=115THENYS=116
1640 IFPEEK(342)=247THENYS=YS+1:
IFYS=173THENYS=172
1650 IFPEEK(343)=247THENXS=XS-1:
IFXS=54THENXS=55
1660 IFPEEK(344)=247THENXS=XS+1:
IFXS=201THENXS=200
1670 RETURN
1680 FORSZ=1TOSB:IFSB$(SZ)=""THE
NNEXT:RETURNELSESX=VAL("&H"+MID$
(SB$(SZ),3,2)):SY=VAL("&H"+RIGHT
$(SB$(SZ),2)):IFXS=SX+3 AND YS=S
Y-1 THEN1690ELSENEXT:RETURN
1690 SOUND240,1:RETURN
1700 IFLA=1THEN1240ELSESOUND120,
1:CIRCLE(XS,YS),6:IFXS=SX+3 AND
YS=SY-1 THENSB$(SZ)="":SS=SS-1:L
INE(XS-3,YS+6)-(XS+3,YS+6),PRESE
T
1710 UZ=U:U=1:B=RND(8)+2:GOSUB40
0:LINE(X6,Y6)-(XS,YS),PSET:U=UZ:
FORX=1TO6:CIRCLE(XS,YS),X:NEXT:F
ORX=6TO1STEP-1:CIRCLE(XS,YS),X,0
:NEXT:POKE65494,0:PLAY"V601L9C":
POKE65495,0:U$="":U=0:GOTO1600
1720 'U.S. LAUNCH ROUTINE
1730 PRINT@480,STRING$(32," "):I
=0:IFLEN(U$)<3THENU$="":GOTO1280
1740 SOUND120,1:U=INSTR(1,U$,"-"
):UL$=RIGHT$(U$,LEN(U$)-U):UR$=L
EFT$(U$,U-1):U=1:U9=1:B=VAL(UL$)
:IFB>0ANDB<11THENGOSUB350ELSEU$=
"":U=0:U9=0:GOTO1280
1750 B=VAL(UR$):IFB>0ANDB<11THEN
GOSUB400ELSEU$="":U=0:U9=0:GOTO1
280
1760 U9=0:IFUS(B)=1THENRETURN
1770 UU=1:YU=Y5:U$="":X=X4:Y=Y4:
X4=X5:Y4=Y5:X5=X:Y5=Y:GOSUB710:U
```

```
=0:RETURN:RETURN
1780 'SUB LAUNCH ROUTINE
1790 IFSS=0THENRETURN
1800 IFJM=1THENPP=30ELSEPP=50
1810 RN=RND(PP):IFRN<>10THENRETU
RN
1820 FORZS=1TOSB:IFSB$(ZS)=""THE
NNEXT:RETURNELSESX=VAL("&H"+MID$
(SB$(ZS),3,2)):SY=VAL("&H"+RIGHT
$(SB$(ZS),2)):SX=SX+5:SY=SY-3
1830 UZ=U:U=1:B=RND(8)+2:GOSUB40
0:LINE(SX,SY)-(X6,Y6),PSET:U=UZ:
IFUS(B)=0THENUS=US-1
1840 US(B)=1:FORX=1TO4:CIRCLE(X6
,Y6),X:NEXT:POKE65494,0:PLAY"V60
1L9C":POKE65495,0:IFUS<1THEN1870
ELSERETURN
1850 'NONEXISTENT SOVIET BASE
1860 RR=0:GOTO340
1870 'NONEXISTANT U.S. BASE
1880 IFU=1THENA$="U.S. BASE"+STR
$(B)+" NO LONGER EXISTS":GOSUB50
:U=0:RETURN
1890 GOTO410
1900 'SOVIET LOSE
1910 PRINT@265,"THEY LOOSE....":
FORX=1TO165:CIRCLE(128,96),X:EXE
C&H549B:IFX=50THEN1920ELSENEXT:G
OTO1930
1920 PRINT@279,"FIRST...":POKE65
494,0:PLAY"V601L9C":POKE65495,0:
NEXT
1930 GOTO1970
1940 'U.S. LOSE
1950 PRINT@266,"YOU LOOSE....":F
ORX=1TO165:CIRCLE(128,96),X:EXEC
&H549B:IFX=50THEN1960ELSENEXT:GO
TO1970
1960 PRINT@279,"FIRST...":POKE65
494,0:PLAY"V601L9C":POKE65495,0:
NEXT
1970 CLS0:A$=" IS IT A GAME, OR
IS IT REAL?  ":GOSUB50:PRINT@257
,A$;:PRINT@288,STRING$(32,CHR$(1
28));
1980 PLAY"V601L9C":FORSZ=1TORND(
300)
1990 I$=INKEY$:IFI$<>""THEN2000E
LSENEXT:GOTO1980
2000 PRINT@480,"";:STOP
2010 'DEFCON DROP
2020 Y=121:X=190
2030 Y=Y+12:IFY>169THENRETURN
2040 GOSUB2050:PAINT(211,Y-11),0
,0:GOTO2030
2050 LINE(210,Y)-(253,Y),PSET:LI
NE-(253,Y+10),PSET:LINE-(210,Y+1
0),PSET:LINE-(210,Y),PSET:LINE(2
34,Y)-(234,Y+10),PSET:PAINT(211,
Y+1),1,1:SOUNDX,1:X=X-10:RETURN
```

CHANGES IN THE MICRO REVOLUTION

By Michael Plog, Ph.D.
Rainbow Contributing Editor

The impact of microcomputers on the educational scene presents several unusual features. Some are downright weird. It is not stretching truth or credibility too much to say that a real revolution is underway. We won't know the full impact of this revolution for a few years yet, but it is easy to tell that changes are happening now. Well, what are some of these features?

First, it is no secret that many students are much more knowledgeable about microcomputers than most teachers. This reversal of roles is completely new in the educational arena. Never before, as far as I know, has a field of study appeared so important to the nation's schools, with the adults in those schools having less knowledge than students.

No matter what you may think about teachers you had (or have), this is definitely not normal. Math teachers have always known more about mathematics than students being taught. The same holds true for teachers of English, the social sciences, and physical sciences. Even in the less traditional courses taught in schools, teachers have almost always started with more knowledge than students.

In many schools around the country, students with microcomputers in their homes were asked to help set up courses and computer labs. Often students provided teacher with a "first lesson" in microcomputers.

Ah, this is just a passing phase of the computer revolution in education; the situation cannot last for very long. Now, notice, I was talking about knowledge of the subject matter. That is different than talking about the ability to learn something, or the intelligence involved. The unbalanced situation of students' knowledge being greater than teachers' will soon pass away — because of the nature of teachers.

(Michael Plog received his Ph.D. degree from the University of Illinois, the M.S. from Memphis State University, and the B.S. from the University of Tennessee. For his foreign/research language option required for the doctorate, he naturally selected computer language. Michael currently works for the Illinois State Board of Education as a research and evaluation specialist.)

Teachers are a strange breed; they return to school themselves, and appear to never tire of learning. After all, education is their way of life, their reason for existence, their base of personal satisfaction and self-image. Have no doubts about it; the next few summers (plus this last one) will see a large number of teachers returning to state universities for courses in computers. I was in a discussion with a man (over 40 years old) talking about his 12-year-old son. The man commented casually, "Jeff took a BASIC course this summer. So did I." All over the country, expect to see teachers pouring over textbooks they would never have noticed a few years ago.

> "There is a different type of person interested in micro applications in schools now than in the past."

This means the students of the future will come to school at age five, knowing less about computers than the teachers. (Teachers do know how to study.) For the present, however, we are stuck with an unusual situation.

There is another passing phase in the computer revolution worth mentioning here. There is a different type of person interested in micro applications in schools now than in the past. Let me tell you a story about a man who is a perfect example of the past type of micro/education. About seven or eight years ago, I worked for a school district. I heard about a sixth grade teacher in the district who had an interest in microcomputers. He even purchases a kit and built a computer. It had few applications, of course, but he was an electronic nut. His kit ended up in his classroom, where the students promptly discovered the power of

humans over machinery. A few years later, I was in a different job, and again heard about this fellow. He now was director of a consortium of schools, with the task of implementing microcomputer education in several districts.

It used to be the case that only one type of person was interested in educational microcomputer applications. We all know thay type of person. Typically male (why?), interested in electronics, wanted to poke around inside machines, enjoyed Adventure games, and was often a member of some futurist organization. When these people buy Color Computers, the first thing they do is take off the case and play with the internal parts. Now, however, the situation is for teachers to learn from; and some may be technicians. It is, however, impossible to replace a teacher in education.

Today, the typical administrator considering putting microcomputers in schools is a lot different from the electronic nut. I have heard several superintendents comment that they did not understand microcomputers, did not like them, never wanted to touch one, even admit they were afraid of computers; but know they had to have the machines. They realized that without some background in microcomputers, their students would be unable to cope in tomorrow's world.

This change of potential educational market may have greater import than we realize. Naturally, the early interest people will continue to expand their microcomputer uses in schools — both in types of applications and number of machines available to students. But the late-comers are more numerous, and represent a different type of market. It would be interesting to see the outcomes of each type of school administrator: attitudes of students toward microcomputers, educational applications of the technology, types of machines purchased, etc. I'm sure someone will do such a study in a few years.

There is one last change, another passing phase, in educational uses of microcomputers to be discussed here. When microcomputers were first being used in school, a common fear was expressed — that teachers would be replaced by machines. Yes, you and I know that is pure bunk, but people who do not understand either education or microcomputers had said it. If anything, the use of microcomputers will mean more educational professionals, not less. Now, some of these professionals may not be teachers. Some will be developers of computer curriculum; some will be resource people for teachers to learn from; and some may be technicians. It is, however, impossible to replace a teacher in education.

Unlike the other two passing phases in the educational computer revolution — type of school people using micros and knowledge level of students compared with adults — this phase is not yet occurring. There is still uncertainty, maybe even fear, in the schools.

A few years from now, this situation will have changed. It is out responsibility to help that change happen. After all, if you have read this far, you must be a computer evangelist. Keep it up, those who will become reluctant converts still need you.

Before I sign off for this month, I want to thank all of you who have written me. I appreciate hearing from you. I'm looking forward to the software evaluation process begun in September. (If you do not know what I'm talking about, dig out your September issue.) It will take several months before anything is reported, but keep mailing those forms.

Until next month, keep thinking education. As long as you keep learning, you may grow up, but you'll never grow old.

Remote Operation Of The Color Computer

By Dan Downard
Rainbow Technical Editor

I guess that the first experience that I had with computers was in school when everyone had to stand in line for hours just to use the keypunch. After you labored over the magic keys for a few hours, you were ready for the acid test. Usually a tray for student projects was situated at the entrance to the computer room. You would place your card deck and the programs would place your card deck and the programs would be run on a first-come, first-served basis, sometimes at night when business use was at a minimum. The first computer I remember was an IBM mainframe with all kinds of flashing lights and switches. It occupied a large room in the basement of the administration building. Other than an operator's console, all input and output was accomplished by punched cards and a printer. My, if they could only see me now.

I guess the only real difference, besides price and state-of-the-art changes, is remote control and time sharing which at that time were very expensive and in their infancy. At the same time, present technology allows home computers with the power of those old mainframes and, with the advent of OS-9, you can now use the CoCo for time sharing and remote operation. In this same issue *the Rainbow* is giving you a bulletin board if you are interested in remote operation of the CoCo. One of the most important aspects of BBS's is the driver routine. This is the link between your computer and the outside world. To be more specific your Serial I/O, or RS-232 port as we like to call it, is your communications link to printers, modems, ham radio, etc.

In this article, we will examine the three major operating systems for the CoCo, Disk BASIC, *FLEX* and *OS-9,* and examine the ways to use remote terminals, whether they be direct connected or by use of a modem. As you are probably aware, *FLEX* and *OS-9* already contain software for remote I/O but what about Disk BASIC? A very short program called *Remote* is included to allow elementary remote control of the CoCo.

In addition to your CoCo, you will need a terminal, or remote computer, whether it be another CoCo, and MC10, Model 100 — anything that has a screen and a keyboard, and a modem. Even the modem is not required if you wish to direct connect the terminal. The remote terminal must have software to translate keystrokes into serial output signals and remote input signals to the screen. Several commercial pieces of software are available such as *Super "Color" Terminal, "Colorcom/E,"* and *"Videotex."* What we are interested in doing is the exact opposite of a terminal. We wish to replace the keyboard with a remote signal and reroute output from the screen to the RS-232 port.

(Dan Downard is an electrical engineer and has been involved in electronics for 24 years through ham radio (K4KWT). His interest in computers began about five years ago and he has built several 68XX systems.)

Disk BASIC

First, several commercial programs are available allowing remote operation of the CoCo. A few that come to mind are *Remoterm* by Star Kits; *RTD3.0,* by Ed March; and *RTD28,* by Lee Blitch. All have many features besides driver routines. Why do you need a program like this in the first place? The three programs mentioned above are synonymous with bulletin boards. As we all know, BBS's are becoming a great medium of information exchange regarding computers. Why not start your own?

On the other hand, there are several other uses for remote operation of the CoCo. Serious word processing requires more than a 32-, 51- or 64-character screen. How about that budget too? How many of us can afford to have two complete computers with disk drives, etc. With a remote terminal, you can have one complete system and one bare-bones terminal to effectively give you two computers. Model 100 owners can have a disk system before they are marketed.

Listing 1 is an elementary remote terminal driver for the CoCo. It allows both local and remote operation of the CoCo. Local I/O is obviously the keyboard and monitor(TV). All output to the screen is also sent at 300 Baud to the Serial I/O port. Input can either be from the keyboard or a 300 Baud signal from the port. No frills or bells and whistles are included in this version. If you have a serious application I would recommend one of the above commercial drivers. *Remote* should provide you with a method of experimentation with a remote terminal.

Running Remote

Remote is written in PIC (position independent code) and

will execute anywhere in memory. To input *Remote* the first time, use a monitor such as *ZBUG* or an assembler such as *EDTASM+*.

With *ZBUG*, enter the object code one byte at a time at your selected address. The listing is assembled at $3F00. A typical session would appear as follows with [E] meaning [ENTER] and [DA] meaning Down Arrow:

```
Z[E]      Enter ZBUG from EDTASM+
B[E]      Enter byte mode
3F00/     Open memory at address $3F00
BE[DA]    Store $BE at $3F00 and increment address
01[DA]    Store $01 at $3F01
68[DA]    Store $68 at $3F02
```

After the program is in memory, save it to tape before you run it using the *ZBUG* P command for the listing given, $3F00 is the start and execute address. Exit *ZBUG* and *CLOADM* the program from BASIC. Type *EXEC* and you are in the remote mode.

FLEX

A remote terminal driver is built-in to Frank Hogg *FLEX 5.0:4*. It is called the EXT command. At any *FLEX* +++ prompt you simply type EXT,[hex 1],. . . ,[hex n] [E] and you have it made. [Hex 1] thru [hex n] are hex numbers sent to change the preset values for terminal parameters such as delay after from feed, delay between characters, etc. A complete description of this command is given in the *Flex* manual.
rate of the external terminal.

A typical remote initialization of *FLEX* would appear as follows:

```
+++SETUP PB300<E>

+++EXT,,,,2,1B,61,2,1B,60<E>
```

The first command sets the Baud rate at 300. The next command enables the remote driver. The hex string 2,1B, 61,2,1B,60 are codes to turn off and on a printer attached to the terminal. This is a very nice feature of *FLEX* since more elaborate terminals also have a printer port.

OS9

I hated to save the best until last, but *OS-9* is *designed for* remote operation. The previous operating systems are capable of remote access with one limitation: Only one program and task can run at a time. Since *OS-9* is both multi-user and multi-tasking, you can now think of your CoCo as a mainframe. While you are using *BASIC09* from the keyboard one of your buddies can run *PASCAL* from a remote terminal. Sounds unreal, doesn't it? Well, it's really simple. *OS9* has three commands that affect remote operation. TSMON is the command that implements time-sharing, or time-sharing monitor. LOGIN provides a security system for logging on the CoCo by remote users. XMODE provides initialization for non standard terminals.

First, let us take a look at the LOGIN command. LOGIN requests a user name and password, which it checks against a validation file named "password." The system notation of the password file is /D0/SYS/password. A typical line in this file would appear as:

```
DAN DOWNARD,REMOTE,1,1,/D0/CMDS,/D1/DP
D,SHELL
```

From left to right the codes stand for user, password, user ID, priority, pathlist of initial execution directory, pathlist of initial data directory and the name of the initial program to execute. In addition to LOGIN, each file in the system has attributes to protect against public access if you so desire.

With XMODE, you can control the characteristics of the remote terminal, or to put it in proper perpective, you can initialize output to the remote terminal to match its parameters. The following parameters are initialized to default values but can be changed if desired:

upc	Upper or upper/lower case
bsb	Erase on backspace
bsl	Backspace over line
echo	Input "echoed" to terminal
lf	Automatic line feed
pause	Pause screen
null	Set null count
pag	Set page length
bsp	Backspace input character
bse	Backspace output character
del	Input delete line character
bell	Bell character
eor	End of record character

eof	End of file character
type	ACIA init. value
reprint	Reprint line character
dup	Duplicate last input line char.
pcs	Pause character
abort	Abort character
quit	Quit character
baud	Set Baud rate

The Baud rate has a range of 110 to 19200. T1 is the device number of the RS-232 port, so a line to initialize the remote terminal at 300 Baud and generate line feeds would appear as:

OS9:XMODE /T1 baud=11 lf

Finally we have TSMON. Assuming you can use default values for the above commands, all you really have to do to activate the timesharing monitor is type "TSMON /T1&." The ampersand (&) is a modifier for concurrent execution. Any XMODE or LOGIN changes should be made before TSMON is activated, but as long as the Baud rate is correct, XMODE can be accessed by the remote user if you have given him access to that command.

All of the *OS-9* commands are well documented in the Radio Shack manual including the above three. At the sake of repeating myself, *OS-9* is designed for remote operation, one of the reasons it is such a popular operating system for small computers.

Hardware

One of the most difficult tasks facing you will be making an interface cable between your CoCo's RS-232 port and the remote terminal or modem. A few hints are in order. For RS-232 operation the serial I/O pinout is defined as follows:

CoCo	Description	RS-232(25 Pin)
Pin 1	Carrier Detect	**
Pin 2	RS-232 IN	Pin 2
Pin 3	Signal Ground	Pin 7
Pin 4	RS-232 OUT	Pin 3

Jumpers may be required on the RS-232 connector. For a start, try shorting Pin 4 to 5 and Pins 6,8, and 20. If this fails, try to obtain assistance from a manual or from the manufacturer.

Summary

Remote operation of the CoCo is a reality for whatever reason you desire. Personally, I have always dreamed of connecting my ham radio to a modem under computer control and carrying on a QSO (contact) from my office at lunch. It's getting closer to reality. I still like to check into bulletin boards and see what's going on, but you might say that *OS-9* will make them obsolete, but who knows? By the way, assuming you have enough disk space to initialize users, *OS-9* will only accept 65535. Any more than that and you're out of luck.

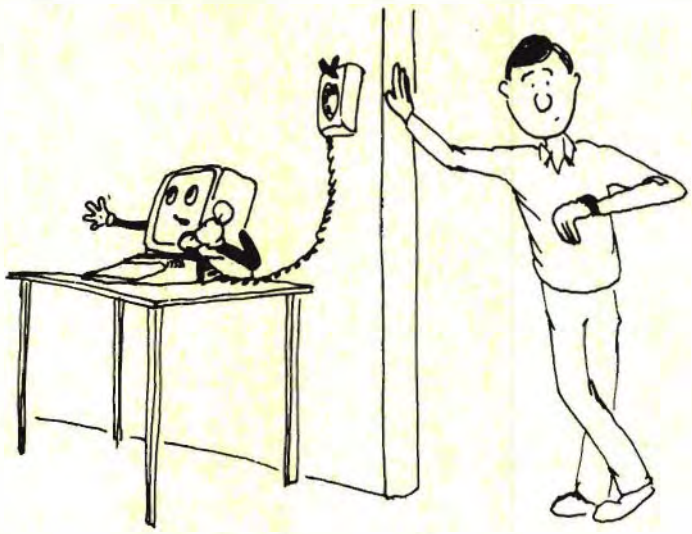
The listing:

```
                        00100 ********************************
                        00110 *           REMOTE             *
                        00120 *A REMOTE TERMINAL DRIVER       *
                        00130 *FOR THE COCO                   *
                        00140 *DAN DOWNARD  RAINBOW  11/83    *
                        00150 ********************************
3F00                    00160         ORG     $3F00
                        00165 *EQUATES FOR ROM AND RAM ADDRESSES
          016A          00170 IHOOK   EQU     $016A
          0167          00180 OHOOK   EQU     $0167
          FF22          00190 PIA     EQU     $FF22
          A000          00200 POLCAT  EQU     $A000
          BE0C          00210 RSOUT   EQU     $BE0C
          00E6          00220 BAUD    EQU     $E6
          00C9          00225 BAUDR   EQU     $C9
          006F          00230 DEV     EQU     $6F
          0070          00240 FLAG    EQU     $70
                        00245 *INITIALIZE RAM HOOKS
3F00 86   C9            00250 START   LDA     #BAUDR
3F02 97   E6            00260         STA     <BAUD
3F04 BE   0168          00270         LDX     1+OHOOK
3F07 AF   8D 0035       00280         STX     1+ORET2,PCR
3F0B BE   0168          00290         LDX     1+IHOOK
3F0E AF   8D 0064       00300         STX     1+IRET2,PCR
3F12 86   7E            00310         LDA     #$7E
3F14 B7   016A          00320         STA     IHOOK
3F17 B7   0167          00330         STA     OHOOK
3F1A 30   8D 000B       00340         LEAX    OUT,PCR
3F1E BF   0168          00350         STX     1+OHOOK
3F21 30   8D 001D       00360         LEAX    IN,PCR
3F25 BF   016B          00370         STX     1+IHOOK
3F28 39                 00380 RET     RTS
```

```
                        00385 *OUTPUT CHARACTER IF DEV=0
                        00386 *INSERT LINE FEEDS TO REMOTE DEVICE
                        00387 *USE ROM SUBROUTINE
3F29 34  02             00390 OUT     PSHS    A
3F2B 0D  6F             00400         TST     <DEV
3F2D 26  0E             00410         BNE     ORET1
3F2F 81  0D             00420         CMPA    #$0D
3F31 26  07             00430         BNE     REMOUT
3F33 86  0A             00440         LDA     #$0A
3F35 BD  8E0C           00450         JSR     RSOUT
3F38 86  0D             00460         LDA     #$0D
3F3A BD  8E0C           00470 REMOUT  JSR     RSOUT
3F3D 35  02             00480 ORET1   PULS    A
3F3F 7E  3F28           00490 ORET2   JMP     RET
                        00495 *INPUT FROM KEYBOARD OR RS-232
                        00496 *IF DEV=0
                        00497 *USE RSIN FOR REMOTE INPUT
3F42 0F  70             00500 IN      CLR     <FLAG
3F44 0D  6F             00510         TST     <DEV
3F46 26  2D             00520         BNE     IRET2
3F48 32  62             00530         LEAS    2,S
3F4A 34  15             00540         PSHS    B,CC,X
3F4C AD  9F A000        00550 IN1     JSR     [POLCAT]
3F50 27  02             00560         BEQ     REMIN
3F52 20  1F             00570         BRA     IRET1
                        00575 *CHECK FOR REMOTE INPUT
3F54 F6  FF22           00580 REMIN   LDB     $FF22
3F57 56                 00590         RORB
3F58 25  F2             00600         BCS     IN1
                        00605 *RS-232 INPUT ROUTINE
                        00606 *BYTE INTO A REGISTER
3F5A 8D  1E             00610 RSIN    BSR     HALF
3F5C C6  01             00620         LDB     #1
3F5E 34  04             00630         PSHS    B
3F60 4F                 00640         CLRA
3F61 8D  15             00650 RSIN1   BSR     FULL
3F63 F6  FF22           00660         LDB     PIA
3F66 56                 00670         RORB
3F67 24  02             00680         BCC     RSIN2
3F69 AA  60             00690         ORA     0,S
3F6B 68  60             00700 RSIN2   ASL     0,S
3F6D 24  F2             00710         BCC     RSIN1
3F6F 32  61             00720         LEAS    1,S
3F71 84  7F             00730         ANDA    #$7F
3F73 35  95             00740 IRET1   PULS    B,CC,X,PC
3F75 7E  3F28           00750 IRET2   JMP     RET
                        00755 *BAUD RATE DELAYS
3F78 8D  00             00760 FULL    BSR     HALF
3F7A 34  02             00770 HALF    PSHS    A
3F7C 96  E6             00780         LDA     <BAUD
3F7E 21  FE             00790 HALF1   BRN     HALF1
3F80 4A                 00800         DECA
3F81 26  FB             00810         BNE     HALF1
3F83 35  82             00820         PULS    PC,A
         3F00           00830         END     START
00000 TOTAL ERRORS
```

Teach CoCo To Dial Up Some Fun

By Ric Manning



There's a device you can plug into your computer that can deliver free software, computer games, loads of technical advice and information and perhaps even a date for Saturday night.

The device is called a modem and when it's matched with a terminal program and your Color Computer, it becomes your passport to a world of fun and information. But be careful. Data-tripping can be addictive and a computer can be like a teenager—once you teach it to use the telephone, it's hard to get it off.

More than 1,000 other computers around the country are out there waiting for yours to dial up for a chat. A growing number of bulletin boards are run on Color Computers and dozens of others are loaded with information, merchandise and fun things to read about and do.

Here are just a few examples:

- The Mines of Moria bulletin board in Houston (713-871-8577) contains 17 tutorials that give you step-by-step instructions for solving some of the most popular adventure games.
- Callers who dial up Dickinson's Movie Guide near Kansas City (913-432-5544) can get a brief review of the movies playing in the Kansas City area along with a list of the stars and a synopsis of the plots.
- The Fantasy Plaza in Burbank, Calif., (213-244-1100) is a computer set up like a large department store. Callers can shop for hardware and software (including CoCo products), videogame cartridges, health and beauty aids and even kitchen products.
- The Big Top Games system in Milwaukee (414-259-9475) lets you play any of about 20 games while you're connected.
- The flagship of the Dial-Your-Match system in Burbank, Calif., (213-842-3322) is an electronic matchmaker on which computer nuts can meet and get to know each other.

Best of all, most of the computer bulletin board systems around the country contain a library of free programs that can be transmitted directly to your computer.

A good example is Dr. D's CoCo Corner run by Gary Dunsford in Pensacola, Fla. (904-456-7195). The system contains 15 programs including a disk utility, copy programs, several games such as *Missile Command* and *Saucer* and programs for creating graphic displays.

Plugging In

So how do you find a seat at this banquet table of computer goodies?

First you need a modem. That's shorthand for modulator-demodulator, a device that translates the digital signal put out by a computer into an analog signal that can be carried by telephone lines. For the Color Computer, any modem that plugs into an RS232 port will do.

Also, you'll need a program that tells your computer to act like a terminal—an extension of the computer you call.

Some of the more popular terminal programs for the Color Computer are Colorcom/E, which is available on disk or ROM pack for about $50, and Nelson's Super Color Terminal, available on tape, disk or ROM pack for $50 to $70. Others include Color Term Plus, Autoterm, DFT II and Videotex.

And, of course, you'll need a telephone. The garden variety phone will do just fine, but one equipped with a Touch-Tone dialer will let you take advantage of long-distance discount services such as Sprint or MCI.

Stepping Out

Once your computer is all dressed up with the right hardware and software, you'll want to find someplace fun for it to visit. You aren't limited to calling only other Color Computers. It makes no difference if the computer you call is an Apple, an Orange, a TRS-80 or even a large mainframe computer; in most instances, you'll be able to speak the same language.

There are several ways of finding other computers to talk to.

* Check The Source and CompuServe. Both publish lists

*(Ric Manning is the editor of **Plumb**, a newsletter that covers news and features about specialty bulletin boards and computer communications services. A five-issue subscription is $20 from **Plumb**, Box 300, Harrods Creek, Ky., 40027.)*

of hundreds of bulletin board systems around the country. The lists, however, are not always up-to-date because systems come and go faster than network television shows. Notices of new boards are often posted in the Color Computer Special Interest Group section.

* Try a specialized newsletter such as PLUMB, which contains news and features about all sorts of computer communications systems plus a list of 100 bulletin boards coded to show their features and special interests.

* Many CoCo boards contain a list of other boards that are run on a CoCo or contain software and technical information of interest to CoCo owners.

* Check the letters section in *the Rainbow*. CoCo owners who start their own bulletin board system often send in letters announcing their boards and listing their phone numbers and hours of operation.

Your modem will know it has reached an available board when it hears a high-pitched tone. Put your computer into terminal mode and the two computers will shake hands and determine if they are speaking the same language.

Most boards will introduce themselves, then ask if you have a password. If you're a new caller, it will ask for your name, the city you're calling from and perhaps your phone number. Passwords are often required to use some of the features, such as the download section. If you plan to be a frequent caller, it's a good idea to ask for one.

Sysop Dru Simon makes callers pass an addition test if they want to enter her "adult" sections. She asks a series of questions designed to prove that callers are 21 years old. If you can't answer her questions, you don't get in. And software-seller Tom Mix says callers have to make five visits to his board before they are allowed to begin downloading software.

Once aboard, the system will show you a set of commands for the functions it supports. One command may adjust the screen width, another will move you from one section to another, and so on.

Here's a sample of some of the commands found on many Color Computer boards:

Application	Lets you apply for a password
Configure	Sets screen display parameters
E-Mail	Lets you send messages to other board users
Help	Just what it says
Merchandise	Catalog shopping section
Time	Tells you how long you've been on the board
View system	Displays information about the board
Board numbers	Displays a list of other boards
Download	Lets you capture programs from the board
Goodbye	Logs you off the system
Logs	Shows a list of the board's regular callers
Sysop page	Calls the board operator for an online chat
Upload	Lets you contribute a program
Want ads	Just like the newpaper
?	Displays the command menu again

If you get confused, try the Help command or page the sysop who can break in, talk directly with you and help you find your way around.

Tech Talk

Many of the early systems were literally bulletin boards—electronic versions of the push-pin-and-paper variety where

notices of club meetings, equipment for sale and help wanted ads were posted.

Ward Christiansen and Randy Suess, two Apple computer and ham radio hobbyists in Chicago, are credited with writing the first BBS software for microcomputers. The board was originally designed as an information clearinghouse for members of the Chicago Area Computer Hobbyist Exchange Club and their CBBS board (312-545-8086) is still populated largely by want ads, meeting notices and requests for technical advice.

Lee F. Blitch said he wrote a BBS program for his Color Computer virtually out of self defense. "I had become a telephone junkie. I was always calling other boards." His board can be examined between 6:00 p.m. and 6:00 a.m. (404-378-4410).

Blitch said he has sold about 15 copies of his BBS program. Interest in telecommunications is relatively new among Color Computer users, he said, "but I think you'll see more and more of it."

Potential sysops, though, should be willing to put in some hard work and a good bit of money. "The amount of work needed to keep a board running is tremendous," said Marshall Goldberg, founder of The Boston Bullet, a TRS-80 board. "The software has to be maintained perfectly and your hardware is running 24 hours a day."

Blitch blamed the demands of the BBS for overheating and his vintage 1980 CoCo.

"I would warn anyone interested in running their own board that if they don't have a lot of time on their hands, they better not get into it," said Greg Moore, who runs a Color Computer board in Arlington, Mass. (617-646-6809).

At least three BBS programs are available for the Color Computer. For details on purchasing them, call Blitch's board; the Silicon Rainbow board in Sunnyvale, Calif., (408-783-6809) or Steve Odneal's board in Kansas City (816-358-6222).

Variety And Spice

Although most boards are stocked with technical advice, some offer their own, unique features. Some may contain stock market reports, movie reviews or online games. Dr. D's board in Pensacola, for example, has a collection of novelty peeks and pokes for the Color Computer and the Doctor's own computer industry rumors and inside dope.

Some systems are also devoted to particular specialties:

* Education-80 in Greenwich, Conn. (203-629-4375) keeps its eye on education conferences around the Northeast and new developments in the use of computers in education.

* The Joke Byte board in Atlantic City (609-927-5922) is loaded with some really awful jokes.

* The Apple-Med board in Iowa City (319-353-6528) contains postings about medical conferences and reviews of medical hardware and software.

* The Magnetic Fantasies board in Los Angeles (213-388-5198) is devoted to discussions about science fiction films, literature and software.

* Want to take a trip to the moon? Check out the Gas-Net board in Greenbelt, Md. (301-344-9156). It's a board that caters to people who want to keep up to date on space flight programs.

* The Notebook in West Palm Beach (305-686-4862) is a place for writers and editors to exchange messages about freelance assignments.

What's Ahead

Several retailers of computer products have found that a BBS is good for business. Moore and Dru Simon said catalog sales of products listed on their boards help pay the board expenses.

And Mix, owner of Tom Mix Software in Grand Rapids, Mich., said his new board (616-364-8217) is a convenience for both his business and its customers. They can call when rates are lowest "and we don't have to answer the phone after five o'clock."

Bob Rosen's business, Spectrum Projects of Woodhaven, N.Y., grew out of Connection-80, a board run on the Model I that was the first board to exclusively serve the Color Computer. Rosen has now added two other boards to his original (212-441-3755).

Rosen and Mix are both working toward the time when a large amount of software will be sold *and delivered* via modem. A couple of other firms are already running such systems. One is The Telephone Software Connection in Torrence, Calif, (213-516-9432) which has developed a system for high-speed transmissions to Apples.

Mix said he believes the day will come when vast amounts of software will be purchased that way. Mix said he's working on ways to prevent rip-offs from such a system.

Others are using bulletin boards for other commercial purposes, such as delivering customized stock reports or special newsletters to members. Howard Young is a former psychologist who explored using a bulletin board to post a newsletter about software for investors.

Young said he thinks the frontiers of electronic information exchange haven't yet been reached. "I think we're at the infant stages of something that will be so explosive that people will wonder why it took us so long to see it."

How To Be A Printer Artist In One Easy Lesson

By Michael J. Himowitz

Ideas for programs can turn up in strange places. In this case, it was the bottom of a cardboard box of 20-year-old junk that I was rescuing from a flooded basement.

There I stumbled upon two books entitled *Typewriter Mystery Games* that I had bought while I was taking a typing course in junior high. They were very simple but, as I dimly remembered, a lot of fun. Each contained sets of instructions for 15 different typewriter pictures: type so many Xs, so many spaces, etc. The gimmick was that you didn't know what you were creating until you were almost through.

Immediately I thought of the computer and the advantages it would have in using this kind of instruction. First, when you're using a typewriter, you often don't know if you've made a mistake until it's too late—then you have to start over from scratch. And even when you have finished a drawing, there's no way to reproduce it, short of using a copier or retyping it. Creating your own drawings is even more difficult for the same reasons.

The computer, on the other hand, gives you the potential for retaining a drawing in memory and allowing you to edit it. Eventually I developed several techniques for doing this, the simplest of which is the program listed below. It is the first step toward developing a library of printer artistry.

Also printed here are three different Printer Mysteries. I won't say much about them, other than they're appropriate for the season. Look for more mysteries in future issues of *the Rainbow*. These, by the way, were created by Julius Nelson, author of many typing textbooks and the founding father of the craft known as "Artyping." They are reprinted with his encouragement and permission.

Here's how to use the program:

When you run *Printer Art* you will get a menu with six choices:

1. CREATE A DRAWING
2. SAVE TO TAPE OR DISK
3. ENTER FROM TAPE OR DISK
4. PRINT DRAWING IN MEMORY
5. EDIT DRAWING

(Michael Himowitz is a Washington correspondent for the Baltimore Evening Sun and proprietor of Federal Hill Software. He uses his computer extensively and has written several programs including "CoCo Accountant." He is interested in meeting people who use their computers in journalism.)

This is what the various options do:

1) CREATE A DRAWING—If you are using the program for the first time, begin with this option. The computer will ask if you want to print out each line you enter as you go along. This is a good way to check on your progress, particularly if you're creating your own drawings. But as you get more familiar with the program, you'll probably want to wait until the end and correct all your mistakes at once.

The computer will then ask you if you want to start a new drawing or continue entering an existing drawing. If you have a drawing in memory, starting a new one will erase everything you have already entered. So, save one drawing to tape or disk before beginning another one.

If you are starting a new drawing, the computer will immediately prompt you for the first line. Select the mystery drawing you wish to enter and look for the first line. We'll use the first line of No. 1 as an example. The instructions are as follows:

44SP 2%

Follow the instructions, type in 44 spaces ("SP" stands for space) and 2 %s. Then hit the ENTER key. Do not put additional spaces at the end of a line. When you hit ENTER, the program will ask you if the line is correct. If it is correct, type "Y" and the computer will prompt you for the next line. If it is not correct, type "N" and the computer will ask you to enter the entire line again. Line 2 consists of 43 spaces and 4 %s, and so on. Other lines include punctuation marks such as colons and periods.

If you are doing your own printer drawings, do not enter lines of more than 80 characters unless you have a printer with a larger carriage and have instructed it to accept lines of more than 80 columns.

The program has two features which will speed up your work. If the line you are entering is identical to the previous line, just type the word SAME and hit the ENTER key when prompted for the line. If the line you are entering is identical to another line earlier in the drawing, just type in the number of the line you wish to duplicate. For example, if you are on line 24 and it is the same as line 12, just enter the number 12.

You may create a drawing up to 80 lines long. When you are finished with the drawing or want to return to the main menu at any time, just hit the ENTER key with no other input when prompted for a line.

This American Eagle is an example of what a drawing will look like but this *is not* one of the three mysteries appearing in this article.

2) SAVE TO TAPE OR DISK—The computer will ask you whether you are using tape or disk (if you are using disk, you still might want to make a tape backup to protect against disk crashes). When you have answered, the computer will ask you for the name of the drawing. Enter a name of no more than seven letters. If you are using disk, do not put an extension on the name, as the program will automatically add the extension "/ART" to your filename.

The program will prompt you to prepare the tape or disk (insert a tape and depress the RECORD and PLAY buttons if you are using a recorder). Then hit ENTER to save the data. When the drawing has been saved, you will be returned to the main menu.

3) ENTER FROM TAPE OR DISK—This is similar to the SAVE function, the only difference being that you should depress only the PLAY button of the recorder. If you are using disk, do not put an extension on the filename. The program will do this for you.

4) PRINT DRAWING IN MEMORY—This is the payoff. The program will ask how many copies you want. Enter the number, then position the paper so the printer head is at the very top and hit the ENTER key. After a brief delay, while the program figures out the centering parameters, you will be told to hit ENTER again. The program will then print out the picture, centered horizontally and vertically.

5) EDIT DRAWING—When you have finished and printed a drawing, you may notice errors or omissions. This function will allow you to correct, delete or add lines to the drawing.

To change a line, use option 1. When prompted, enter the number of the line you wish to change. The computer will print that line on the screen and prompt you for a new one. To delete the line, just type in DEL and hit the ENTER key. To change it, type the whole line again (correctly this time) and hit the ENTER key. At this point, the computer will ask if you want to change another line. If you do, type "Y" and the computer will repeat the process. If not, type "N" and you will return to the main menu.

To insert a line, use option 2. The computer will ask you where you want the line inserted. Enter the number of the line immediately preceding the line you wish to insert. For example, if you want to insert a line after line 23, ENTER the number 23. Then type in the new line and hit the ENTER key. Here the program will ask you if the new line you've entered is correct. If it is not, the computer will go back to the start of the process, which means you must again type in the number of the line after which you want the new line inserted.

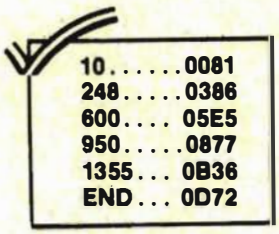
Odds 'n Ends

The program will work with any printer. If you have a printer with different fonts, you might want to experiment by typing in directly the command for the font you want before loading the program. It will not, however, work with expanded 10 cpi fonts. Also, use of the condensed font available with Epson, Gemini and Okidata printers will result in a somewhat distorted picture because the type itself is more elongated than the standard size.

I have also encountered a problem of some columns not lining up because my printer is set to receive at 2400 Baud (standard is 600 Baud). If you are running this fast and have trouble, you might want to poke in a longer line printer delay (*POKE 151,128:POKE 152,0*) to straighten things out.

If you look carefully at the program, you may notice that it converts leading spaces into "<" marks and colons into "=" signs. This is because I/O routines have trouble with these characters. The program converts them back to the proper characters before printing.

For those with a further interest in the subject, a complete four-program cassette and tutorial with 12 ready-to-run pictures and instructions for 49 more are available for $19.95 from Federal Hill Software, 825 William St., Baltimore, Md., 21230.



The listing:

```
1 ' PRINTER ARTIST
2 ' (C) 1983 BY MICHAEL J. HIMOW
ITZ
3 ' FOR PERSONAL USE OF RAINBOW
READERS ONLY
5 PCLEAR1
10 CLEAR 5000
20 DIM L$(80)
25 A=1:NN=-2
30 CLS:PRINT "          PRINTER A
RTIST":PRINT:PRINT:PRINT"1. CREA
TE A DRAWING":PRINT"2. SAVE TO T
APE OR DISK":PRINT"3. ENTER FROM
 TAPE OR DISK":PRINT"4. PRINT DR
AWING IN MEMORY":PRINT"5. EDIT D
RAWING"
35 IF NF$="" THEN 40 ELSE PRINT@
480-32,"current file: "+NF$
40 CH$=INKEY$:IF CH$="" THEN 40
50 ON VAL(CH$) GOTO 200, 400, 60
0, 800, 1000
60 GOTO 40
200 CLS:PRINT"   CREATE A DRAWIN
G":GOSUB 15300
203 PRINT:PRINT"1. START A NEW D
RAWING":PRINT"2. CONTINUE CURREN
T DRAWING"
204 HG$=INKEY$:IF HG$="" THEN 20
```

```
4
205 IF VAL(HG$)=1 THEN A=1
220 FOR X=A TO 80
230 PRINT"LINE NO."X:LINE INPUT
L$(X)
233 IF L$(X)="SAME" THEN L$(X)=L
$(X-1)
234 V=VAL(L$(X)):IF V>0 THEN L$(
X)=L$(V)
235 IF PL=1 AND L$(X)<>"" THEN P
RINT#-2,L$(X)
240 IF L$(X)="" THEN 350
243 GOSUB 245:GOTO 260
245 IF MID$(L$(X),1,1)=" " THEN
MID$(L$(X),1,1)="<"
247 FOR B=1 TO LEN(L$(X)):IF MID
$(L$(X),B,1)=":" THEN MID$(L$(X)
,B,1)="="
248 NEXT B
249 RETURN
260 PRINT"IS THIS CORRECT? (Y/N)
"
270 CR$=INKEY$:IF CR$="" THEN 27
0
280 IF CR$<>"Y" THEN PRINT"CORRE
CTED":GOTO 230
290 NEXT X
300 GOTO 30
350 X=X-1:A=X+1:GOTO 30
```

```
400 CLS:PRINT"  SAVE DRAWING":PR
INT:PRINT"TAPE (T) OR DISK (D)?"
401 GOSUB 405
402 INPUT "NAME OF DRAWING";NF$:
NF$=LEFT$(NF$,7)
403 GOTO 430
405 MD$=INKEY$:IF MD$="" THEN 40
5
410 IF MD$="T" THEN MD=-1 ELSE I
F MD$="D" THEN MD=1 ELSE GOTO 40
5
420 IF MD=1 THEN ZR$="DISK" ELSE
 IF MD=-1 THEN ZR$="TAPE"
425 RETURN
430 PRINT"PREPARE "+ZR$:INPUT "A
ND PRESS enter";PE
440 PRINT"SAVING "+NF$+" TO "ZR$
450 IF MD=1 THEN IF RIGHT$(NF$,3
)<>"DAT" THEN NF$=NF$+"/ART"
460 OPEN "O", #MD, NF$
470 PRINT#MD,X:FOR K=1 TO X:PRIN
T#MD,L$(K):NEXT K:CLOSE#MD:GOTO
30
600 CLS:PRINT"ENTER FROM TAPE OR
 DISK"
605 PRINT"TAPE (T) OR DISK (D)?"
610 GOSUB 405:INPUT"NAME OF DRAW
ING";NF$:NF$=LEFT$(NF$,7):IF MD=
1 THEN NF$=NF$+"/ART"
```

```
615 PRINT"PREPARE "+ZR$:INPUT "A
ND PRESS enter";PE
620 PRINT"LOADING "+NF$+" FROM "
ZR$"
630 OPEN "I",#MD,NF$: INPUT#MD,X
:FOR K=1 TO X:LINE INPUT#MD,L$(K
):NEXT K:CLOSE#MD:GOTO 30
800 CLS:PRINT"PRINT "+NF$:PRINT:
INPUT "HOW MANY COPIES";BB
810 PRINT"POSITION TOP OF PAPER"
:PRINT"AT PRINTER HEAD":INPUT "A
ND PRESS enter";PE
815 GOSUB 2000
820 FOR F=1 TO BB:CLS:PRINT NF$+
" COPY NO."F
830 KK=66-X:AA=INT(KK/2)
840 FOR R=1 TO AA:PRINT#NN:NEXT
R
850 FOR K=1 TO X
860 IF MID$(L$(K),1,1)="<" THEN
MID$(L$(K),1,1)=" "
870 FOR B=1 TO LEN(L$(K)):IF MID
$(L$(K),B,1)="=" THEN MID$(L$(K)
,B,1)=":"
880 NEXT B
930 PRINT#-2,TAB(MR)L$(K)
935 FOR W=1 TO 30:NEXT W
940 NEXT K
945 FOR F=1 TO AA:PRINT#NN:NEXT
950 PRINT#NN
960 GOTO 30
1000 CLS:PRINT"  EDIT A LINE":PR
INT:PRINT"1. CHANGE A LINE":PRIN
T"2. INSERT A LINE"
1030 Q$=INKEY$:IF Q$="" THEN 103
0
1040 ON VAL(Q$) GOTO 1100,1300
1050 GOTO 1000
1100 CLS:PRINT"CHANGE A LINE":PR
INT:INPUT "WHICH LINE?";WN
1120 PRINT"CURRENT LINE":PRINTL$
(WN)
1130 PRINT"ENTER CORRECTED LINE"
:LINE INPUT J$
1140 IF J$="DEL" THEN 1150 ELSE
1200
1150 FOR K=WN TO X-1:L$(K)=L$(K+
1):NEXT K:X=X-1:A=X+1:GOTO 1280
1200 L$(WN)=J$:GOSUB 1400
1280 PRINT"ANOTHER CHANGE?"
1283 AC$=INKEY$:IF AC$="" THEN 1
283
1285 IF AC$="Y" THEN 1100 ELSE 3
0
1300 CLS:PRINT"INSERT A LINE":PR
INT:PRINT"LINE AFTER WHICH":INPU
T "NEW LINE WILL BE INSERTED";WN
1310 PRINT"ENTER NEW LINE":LINE
INPUT J$
1315 PRINT:PRINT"IS THIS CORRECT
?"
1320 RC$=INKEY$:IF RC$="" THEN 1
320
1325 IF RC$<>"Y" THEN 1300
1330 IL=WN+1
1340 FOR K=X+1 TO IL+1  STEP-1:L
$(K)=L$(K-1):NEXT K:L$(IL)=J$
1350 GOSUB 1500
1355 X=X+1:A=X+1
1360 GOTO 30
1400 IF MID$(L$(WN),1,1)=" " THE
N MID$(L$(WN),1,1)="<"
1410 FOR B= 1 TO LEN(L$(WN))
1420 IF MID$(L$(WN),B,1)=":" THE
N MID$(L$(WN),B,1)="="
1430 NEXT B
1440 RETURN
1500 IF MID$(L$(IL),1,1)=" " THE
N MID$(L$(IL),1,1)="<"
1510 FOR B=1 TO LEN(L$(IL)):IF M
ID$(L$(IL),B,1)=":" THEN MID$(L$
(IL),B,1)="="
1520 NEXT B
1530 RETURN
2000 MJ=0
2010 FOR K=2 TO X
2020 IF LEN(L$(K))=>LEN(L$(K-1))
 THEN MJ=LEN(L$(K)):MR=INT((80-M
J)/2)
```

```
2030 NEXT K
2040 IF MJ=>80 THEN MR=0
2050 RETURN
2060 FOR K=1 TO X
2070 PRINT#-2, USING "###";K;:PR
INT#-2, "   "+L$(K)
2080 NEXT K:GOTO 25
15300 PRINT:PRINT"DO YOU WANT TO
 PRINT":PRINT"LINES WHEN ENTERED
? (Y/N)"
15310 PL$=INKEY$:IF PL$="" THEN
15310
15320 IF PL$="Y" THEN PL=1 ELSE
PL=0
15330 RETURN
```

Printer Mystery 1

LINE NO.
1 — 44sp 2%
2 — 43sp 4%
3 — 43sp 5%
4 — 44sp 6%
5 — 45sp 5%
6 — 45sp 6%
7 — 46sp 6%
8 — 46sp 7% 7sp 2%
9 — 45sp 9% 5sp 3%
10 — 45sp 16%
11 — 45sp 16%
12 — 45sp 14%
13 — 42sp 17%
14 — 40sp 20%
15 — 40sp 21%
16 — 43sp 16%
17 — 34sp 26%
18 — 31sp 28%
19 — 28sp 28%
20 — 26sp 30%
21 — 24sp 32%
22 — 22sp 34%
23 — 20sp 36%
24 — 18sp 39%
25 — 16sp 43%
26 — 15sp 46%
27 — 14sp 54%
28 — 10sp 65% 3sp 2%
29 — 5sp 74%
30 — 79%
31 — 1sp 77%
32 — 2sp 23% 3sp 45%
33 — 8sp 14% 6sp 44%
34 — 28sp 36% 3sp 4%
35 — 28sp 37%
36 — 28sp 36%
37 — 29sp 34%
38 — 30sp 32%
39 — 30sp 32%
40 — 32sp 30%
41 — 37sp 24%
42 — 29sp 2% 2sp 27%
43 — 25sp 34%
44 — 14sp 5% 1sp 12% 2sp 24%
45 — 8sp 23% 2sp 24% 9sp 3%
46 — 28% 3sp 26% 3sp 10%
47 — 25% 4sp 40%
48 — 1sp 21% 4sp 40%
49 — 3sp 15% 4sp 39%
50 — 3sp 14% 4sp 24%
51 — 4sp 9% 4sp 23%
52 — 6sp 5% 3sp 23%
53 — 6sp 3% 3sp 22%
54 — 6sp 2% 4sp 16%
55 — 14sp 8%

Halloween Printer Surprise

The following are instructions, *not* BASIC listings. For instance, line one instructs you to hit the [SPACEBAR] 44 times and the percent key two times.

Printer Mystery 2

LINE 1: 24SP 3X 5SP 5X
LINE 2: 23SP 1X 1SP 1X 1SP 1X 4SP 1X 2SP 5X
LINE 3: 22SP 1X 2SP 1X 2SP 1X 3SP 1X
LINE 4: 21SP 1X 3SP 1X 3SP 1X 2SP 1X
LINE 5: 20SP 1X 4SP 1X 4SP 1X 1SP 1X
LINE 6: 19SP 1X 3SP 5X 2SP 5X
LINE 7: 13SP 1X 4SP 1X 4SP 5X 2SP 5X
LINE 8: 13SP 1X 3SP 1X 5SP 5X 2SP 5X
LINE 9: 13SP 1X 2SP 1X 6SP 5X 2SP 5X
LINE 10: 11SP 5X 7SP 5X 2SP 7X
LINE 11: 10SP 6X 9SP 1X 6SP 1X 3SP 2X
LINE 12: 1X 8SP 7X 7SP 5X 2SP 5X 2SP 2X
LINE 13: 1SP 2X 5SP 8X 7SP 5X 2SP 5X 3SP 2X
LINE 14: 2SP 2X 3SP 9X 7SP 5X 2SP 5X 3SP 5X
LINE 15: 3SP 4X 16SP 5X 2SP 5X 3SP 6X
LINE 16: 3SP 13X 7SP 5X 2SP 5X 3SP 7X
LINE 17: 4SP 16X 5SP 1X 6SP 1X 5SP 8X
LINE 18: 5SP 28X 2SP 8X
LINE 19: 7SP 34X
LINE 20: 10SP 29X
LINE 21: 11SP 26X
LINE 22: 11SP 24X

Printer Mystery 3

LINE 1: 10SP 3U 2SP 3U 3SP 2U 1SP 3U 1SP 3U 1SP 2U 3SP 3U 2SP 3U
LINE 2: 8SP 13U 2: 9U 2: 13U
LINE 3: 8SP 14U 2: 7U 2: 14U
LINE 4: 8SP 15U 2: 5U 2: 15U
LINE 5: 8SP 16U 2: 3U 2: 16U
LINE 6: 8SP 16U 1: 5$ 1: 16U
LINE 7: 7SP 12U 4SP 9$ 4SP 12U
LINE 8: 3SP 13U 5SP 13$ 5SP 13U
LINE 9: 3SP 12U 2SP 21$ 2SP 12U
LINE 10: 3SP 5U 9SP 21$ 9SP 5U
LINE 11: 3SP 4U 9SP 23$ 9SP 4U
LINE 12: 3SP 4U 9SP 23$ 9SP 4U
LINE 13: 3SP 4U 9SP 23$ 9SP 4U
LINE 14: 3SP 4U 8SP 12$ 1I 12$ 8SP 4U
LINE 15: 3SP 4U 8SP 12$ 1I 12$ 8SP 4U
LINE 16: 3SP 4U 7SP 13$ 1I 13$ 7SP 4U
LINE 17: 3SP 4U 5SP 15$ 1I 15$ 5SP 4U
LINE 18: 1SP 7U 2SP 18$ 1I 16$ 2SP 7U
LINE 19: 1SP 7U 1SP 20$ 1I 16$ 1SP 7U
LINE 20: 1SP 7U 1SP 21$ 1I 15$ 1SP 7U
LINE 21: 1SP 3U 1SP 3U 17SP 5$ 17SP 3U 1SP 3U
LINE 22: 1SP 3U 1SP 3U 18SP 3$ 18SP 3U 1SP 3U
LINE 23: 1SP 3U 1SP 5U 35SP 5U 1SP 3U
LINE 24: 1SP 53U
LINE 25: 55U
LINE 26: 55U

Design Your Own Champion

By Bob Albrecht
Rainbow Contributing Editor

The Carefully Contrived Superhero

In the game of *Champions,** you create a superhero like Batman or Wonder Woman or Spiderman. Your character has eight basic characteristics. Each characteristic has a base value of 10. You can increase the value of a characteristic by spending Power Points according to the following price schedule.

CHARACTERISTIC	ABBREVIATION	PRICE
Strength	STR	1
Dexterity	DEX	3
Constitution	CON	2
Body Pips	BOD	2
Intelligence	INT	1
Ego	EGO	2
Presence	PRE	1
Comeliness	COM	.5

We have written a simple "worksheet" program to help a player design a *Champions* character. When your type *RUN*, it begins like this:

	CHARACTERISTIC	VALUE	PRICE	POINTS
1	STR	0	1	0
2	DEX	0	3	0
3	CON	0	2	0
4	BOD	0	2	0
5	INT	0	1	0
6	EGO	0	1	0
7	PRE	0	1	0
8	COM	0	.5	0

TOTAL POINTS: 0
YOU CAN CHANGE ANY VALUE OR
PRESS ZERO (0) TO START OVER

CHANGE (1 TO 8)?

(Bob Albrecht and George Firedrake are two of the most prolific authors in the microcomputer world today. Specialists in writing for beginners, they are authors of numerous books, including TRS-80 Color Basic.)

*For information about *Champions,* contact Hero Games, 92A 21st Avenue, San Mateo, Calif., 94402.

Just like it says, press ZERO (0) to start over or press a number from 1 to 8 to change the value of a characteristic. If you press a number key from 1 to 8, the CoCo asks:

NEW VALUE?

Type your new value and press <ENTER>. The new value appears on-screen along with a new TOTAL POINTS. You can select a characteristic and enter a new value as many times as you want until you get the character you want, with exactly the number of TOTAL POINTS the GM said you could have.

We begin by setting up fixed arrays to hold the characteristic abbreviations (CH$) and the prices (PRICE).

```
100 REM**CONTRIVE A SUPERHERO
199 '
200 REM**SET UP FIXED ARRAYS
210 FOR K=1 TO 8
220 : READ CH$(K), PRICE(K)
230 NEXT K
240 DATA STR,1,DEX,3,CON,2,BOD,2
,INT,1,EGO,1,PRE,1,COM,.5
```

Next, we initialize the variable information by setting the characteristic values (CVAL), points spent for each characteristic (PTS), and total points spent (TTL) to zero.

```
300 REM**SET VARIABLE INFO TO 0
310 FOR K=1 TO 8
320 : CVAL(K) = 0
330 : PTS(K) = 0
340 NEXT K
350 TTL = 0
399 '
```

Inside the CoCo, everything is set up and ready to go. Let's put it on the screen.

```
400 REM**PRINT HEADINGS
410 CLS
```

```
420 PRINT "CHARACT." TAB(10) "VA
LUE" TAB(17) "PRICE" TAB(24) "PO
INTS"
500 REM**PRINT WORKSHEET DATA
510 FOR K=1 TO 8
520 : PRINT K TAB(4) CH$(K);
530 : PRINT TAB(10) CVAL(K);
540 : PRINT TAB(17) PRICE(K);
550 : PRINT TAB(24) PTS(K)
560 NEXT K
570 PRINT TAB(10) "TOTAL POINTS:
" TTL
```

The information is on-screen. Find out what the user wants to do.

```
600 REM**CHANGE WHAT VALUE?
610 PRINT
620 PRINT "YOU CAN CHANGE ANY VA
LUE OR"
630 PRINT "PRESS ZERO (0) TO STA
RT OVER"
640 PRINT
650 PRINT "CHANGE (1 TO 8)?";
700 REM**GET USER'S RESPONSE
710 K$=INKEY$:IF K$="" THEN 710
720 IF K$="0" THEN 310
730 IF K$<"1" THEN 710
740 IF K$>"8" THEN 710
750 CN = VAL(K$)
760 PRINT CN
```

Line 710 causes the CoCo to scan the keyboard until someone presses a key. If you press the zero (0) key, line 710 sends the CoCo back to line 310 to start over. Lines 720 and 730 cause the CoCo to ignore any key other than a number key, 1 to 8. Line 750 converts the string value of K$ to a numeric value.

Next, find out the new value for the characteristic to be changed.

```
800 REM**GET NEW VALUE
810 INPUT "NEW VALUE"; CVAL(CN)
```

The new value replaces the old value of the characteristic in the array CVAL. Of course, this changes the points for the characteristic and the total points for all eight characteristics. So let's now compute these values, then loop back for another change.

```
900 REM**COMPUTE POINTS AND TTL
910 TTL = 0
920 FOR K=1 TO 8
930 : PTS(K) = CVAL(K)*PRICE(K)
940 : TTL = TTL + PTS(K)
950 NEXT K
1000 REM::GO AROUND AGAIN
1010 GOTO 410
```


Things To Come

Next time we will start writing programs to store information on cassette tapes. If you don't know how to set up and use cassette files, we suggest you read pages 219 through 230 in "Getting Started With Color BASIC." Try your hand at writing these programs:

- The Name Machine #1. Write a program to ask for a name structure (See "GameMaster's Apprentice," July, 1983), then generate random names, one at a time. Press "S" to save the name on cassette tape. Press the space bar for another name. Press "N" to enter a new structure.
- The Name Machine #2. Write a program to ask for a name structure, then generate names numbered zero through nine. You can save any name to tape by pressing its number key. Press the space bar for ten more names. Press "N" to enter a new structure.
- The Name Machine #3. Write a program to select name structures at random from a list of name structures (perhaps in *DATA* statements or in an array). Generate ten names numbered zero through nine. You can save any name to tape by pressing its number key. Press the space bar for ten more names.
- Store Character Records On The Tape. Write a program to store the name and seven basic characteristics for a character on a cassette tape. Enter the information from the keyboard in response to questions on the screen.
- Find Character Record. Write a program to search a cassette file for a character record by name of character. If the record is found, display it. If it is not found, print an appropriate message.
- Scan Character File. Write a program to read and display all records in a cassette file, one by one. Press the space

bar to get the next record.

•Load Character Arrays. Write a program to read all character records in a cassette file into arrays NAYM$, STR, CON, SIZ, INT, POW, DEX, and CHA.

Well, that should be enough to keep you busy until the next issue of *the Rainbow* arrives.

Clubs and Cons

How do you learn about fantasy role-playing games? How do you find people to play with? Easy. Join a club or go to a convention. Here is a brief list of clubs and people who run conventions:

Drangonsteeth
A & J Hobby House
20 Auburn Ave.
Utica, N.Y. 13501

Dragon Hunters' Guild
c/o Keith Payne
Rt. 1, Box 794
Winfield, W.Va. 25213

Unicorn Adventurers
P.O. Box 12666
Lexington, Ky. 40583

Forest Gamers Club
River Forest Community Center
414 Jackson
River Forest, Ill. 60305

Society Of Wizards & Warriors
P.O. Box 168
Julian A. McPhee U.U.
Cal Poly
San Luis Obispo, Calif. 93407

Eastcon
P.O. Box 139
Middletown, N.J. 07748

X-Con
P.O. Box 7
Milwaukee, Wis. 53201

Denver Gamers Association
P.O. Box 2945
Littleton, Colo. 80161

Kommander's Wargaming Club
P.O. Box 2235
Mansfield, Ohio 44905

Gamemasters Guild
1413 Washington St.
Waukegan, Ill. 60085

Seaga
P.O. Box
Norcross, Ga. 30093

Mythopoeic
P.O. Box 711
Seal Beach, Calif. 90740

Omacon
2518 S. 167th St.
Omaha, Neb. 68130

Nancon
118 Briargrove Center
6100 Westheimer
Houston, Texas 77057

Memphis Fantasy Con
665 S. Highland
Memphis, Tenn. 38111

Archon
P.O. Box 15852
Overland, Mo. 63114

Mysticon
P.O. Box 1367
Salem, Va. 24153

Texakron
1021 East 29th
Texarkana, Ariz. 75502

Conquest
P.O. Box 36212
Kansas City, Mo. 64111

Marcon
P.O. Box 2583
Columbus, Ohio 43216

Dallcon
P.O. Box 345125
Dallas, Texas 75230

Metro Detroit Gamers
2616 Kenwyck
Troy, Mich. 48098

Gateway
Strategicon PR Dept.
P.O. Box 2577
Anaheim, Calif. 92804

Grimcon
P.O. Box 4153
Berkeley, Calif. 94794

Cincinnati Adventure Gamers
11020 Reading Road, Suite 175
Sharonville, Ohio 45241

OSU Wargamers
700 West Scott #321
Stillwater, Okla. 74074

Vikingcon
Associated Students
Viking Union 402
Western Washington U
Bellingham, Wash. 98225

If we didn't mention your club or con, let us know about it. George and Bob, P.O. Box 310, Menlo Park, CA 94025.

The listing:

```
399.....0117
630.... 028B
END....0418
```

```
100 REM**CONTRIVE A SUPERHERO
199 '
200 REM**SET UP FIXED ARRAYS
210 FOR K=1 TO 8
220 : READ CH$(K), PRICE(K)
230 NEXT K
240 DATA STR,1,DEX,3,CON,2,BOD,2
,INT,1,EGO,1,PRE,1,COM,.5
299 '
300 REM**SET VARIABLE INFO TO 0
310 FOR K=1 TO 8
320 : CVAL(K) = 0
330 : PTS(K) = 0
340 NEXT K
350 TTL = 0
399 '
400 REM**PRINT HEADINGS
410 CLS
420 PRINT "CHARACT." TAB(10) "VA
LUE" TAB(17) "PRICE" TAB(24) "PO
INTS"
499 '
500 REM**PRINT WORKSHEET DATA
510 FOR K=1 TO 8
```

```
520 : PRINT K TAB(4) CH$(K);
530 : PRINT TAB(10) CVAL(K);
540 : PRINT TAB(17) PRICE(K);
550 : PRINT TAB(24) PTS(K)
560 NEXT K
570 PRINT TAB(10) "TOTAL POINTS:
" TTL
599 '
600 REM**CHANGE WHAT VALUE?
610 PRINT
620 PRINT "YOU CAN CHANGE ANY VA
LUE OR"
630 PRINT "PRESS ZERO (0) TO STA
RT OVER"
640 PRINT
650 PRINT "CHANGE (1 TO 8)?";
699 '
700 REM**GET USER'S RESPONSE
710 K$=INKEY$:IF K$="" THEN 710
720 IF K$="0" THEN 310
730 IF K$<"1" THEN 710
740 IF K$>"8" THEN 710
750 CN = VAL(K$)
760 PRINT CN
799 '
800 REM**GET NEW VALUE
810 INPUT "NEW VALUE";CVAL(CN)
899 '
900 REM**COMPUTE POINTS AND TTL
910 TTL = 0
920 FOR K=1 TO 8
930 : PTS(K) = CVAL(K)*PRICE(K)
940 : TTL = TTL + PTS(K)
950 NEXT K
999 '
1000 REM::GO AROUND AGAIN
1010 GOTO 410
```


FANTASY ROLE PLAYING GAMES

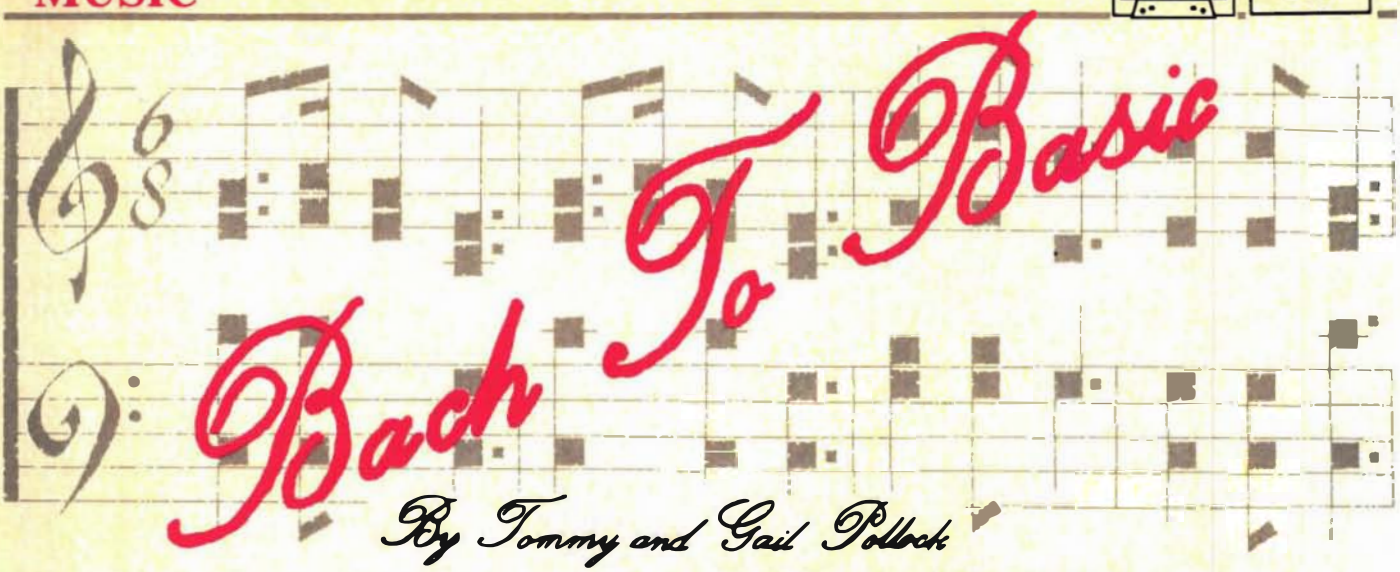
Millions of young people, and many not-so-young, are playing fantasy role playing games. A role playing game is a game in which one or more players create and control characters (adventurers) who live their imaginary lives in a specially made game world. The game world is created, managed, and operated by a GameMaster (GM), also called a referee, adventure master, or dungeon master (DM).

Most people who play role playing games use a formal rule system. Some of the best known are shown below.

Dungeons & Dragons (D&D). *From TSR Hobbies, P.O. Box 756, Lake Geneva, WI 53147.*
RuneQuest (RQ). *From Chaosium, P.O. Box 6302, Albany, CA 94706.*
Tunnels & Trolls (T&T). *From Blade, Box 1467, Scottsdale, AZ 85252.*
Worlds of Wonder (WOW). *From Chaosium, P.O. Box 6302, Albany, CA 94706.*

BEGINNERS BEWARE! The rule books are very difficult to understand. If you are a beginner, first try ***Worlds of Wonder*** *or* ***Tunnels & Trolls.*** *Programs in "GameMaster's Apprentice" are based on the game system used in* ***Worlds of Wonder*** *and* ***RuneQuest.*** *For general information about fantasy role playing games, try the following book, excellent for beginners.*

Through Dungeons Deep *by Robert Plamondon. From Reston Publishing Company, 11400 Sunset Hills Road, Reston, VA 22090.*



Bach To Basic

By Tommy and Gail Pollock

Can't read a note? Spent all your money on computers and can't afford a stereo? If so, this program will solve your problems. After loading this program, woo your friends and relatives with your musical ability as your computer plays "Sinfonia" by J. S. Bach.

CoCo-Bach requires 16K and Extended Color BASIC. After *CLOADing* "Sinfonia," type *RUN*, and after a title you will be asked to press a key to begin. At the end of the music, pressing any key will repeat the program.

```
40 ......01FC
80 ..... 03EC
END ..... 05E9
```

The listing:

```
1 GOSUB 1000
2 FOR X=1 TO 100:NEXT X
10 PLAY"O2L8CO5L16CO4BO5L8CO4GEG
L16CDCO3BO4L8CO3GEGL16CGDGEGFGEG
DGCO4CO3BAGO4CO3BAGFEDCGDGEGFGEG
DGCO4CO3BAGO4CO3BAGFEDCDEFGABO4C
DEFDEGCDEFGABO5CO4ABO5CO4GFGEGFG
EGDG"
20 PLAY"CO5CO4BAGCACGCFC EGDGEGF
GEGDGCO5CO4BAGCACGCFC EO3CO4CO3C
BCO4CO3"
30 PLAY"CO4DO3CBC O4CO3CO4EO3CO4
DO3CO4EO3CO4FO3CO4DO3C O4ECO3CO4
CO2BO4CO3CO4CO3DO4CO2BO4C O3CO4C
O3EO4CO3DO4CO3EO4CO3F"
40 PLAY"O4CO3DO4C O3CDCDEGCDEGCD
 EFEFGO4CO3EFGO4CO3EF GAGAB-O4EO
3GAB-O4EO3GA B-O4GECO3B-GECO2B-A
B-G ABABO3C#EO2ABO3C#EO2AB"
50 PLAY"O3C#DC#DEAC#DEAC#D EFEFG
O4C#O3EFGO4C#O3EF GO4EC#O3AO4GEF
DC#EO3AG FAFDO4DO3BO4CO3AG#BED C
ECO2AO3CEAEO4CO3AO4EO3A"
60 PLAY"G#BG#EO4ED#ED#EO3BO4CO3A
 G#BG#EO4DC#DC#DO3BO4CO3A G#BG#E
AEG#EL8AF L4EO4DP8L8O3B L4O4CO3A
P8O4L8D# L4EO5DP8L8O4B L4O5CO4AP
8L8O3A"
70 PLAY"L16 CD#AD#O4CO3D#AD#O4CO
3D#AD# CD#AD#O4CO3D#AD#O4CO3D#AD
# O2BO3EAEBEAEBEAE DEG#EBEG#EBEG
#E AO4AGFEAEDCECO3B AO4AEDCECO3B
AO4CO3AG L8F#DF#AL4O4C L16O3BO4G
F#EDGDCO3BO4DO3BA GO4GFEDFDCO3BO
4DO3BA"
80 PLAY"L2O4FL16FEFD ECEGO5CO4GE
CP16O4GO5CO4B"
90 PLAY"L4O5CO4L2G L8GL16FEL2F L
8FL16EDL4EL16ECO3GO4C O3FO4CDCO3
BGBO4CDO3GO4DE L4.FL8GL4E L16EFE
L32FDL32DEDEDEDEDEDEDEL4.DL8C
L16CO5CO4BAGO5CO4FO5CO4EO5CO4DO5
C O4CO5CO4BAGCFCECDC P16O3EGBO4C
EGBL4O5C"
900 PRINT@394,"PRESS ANY KEY"
910 I$=INKEY$:IF I$=""THEN 910 E
LSE 1
999 END
1000 CLS:PRINT@43,"SINFONIA"
1010 PRINT@73,"BY J.S. BACH"
1020 PRINT@139,"ARRANGED"
1030 PRINT@165,"FOR THE COLOR CO
MPUTER"
1040 PRINT@206,"BY"
1050 PRINT@233,"TOMMY POLLOCK"
1060 PRINT@269,"AND"
1070 PRINT@298,"GAIL POLLOCK"
1080 PRINT@357,"PRESS ANY KEY TO
 BEGIN"
1090 I$=INKEY$:IF I$="" THEN 109
0 ELSE RETURN
```

(Tommy Pollock is an eighth grader, a little leaguer, and was a drummer in his school band. His mother, Gail Pollock, is a published composer and an instructor of business and economics at Gordon Junior College.)

PIPELINE

WHO KNOWS WHAT'S for the CoCo? Already, Radio Shack has given us one of the most sophisticated operating systems on the market in OS-9, and now, this month an Ink Jet printer will really liven things up for the Color Computer.

But, truth be it known, there are more and more people — including those in Fort Worth, as we hear it —working on loads and loads of other things for CoCo in the months and years ahead.

A year ago we wrote that a full-blown compiler would be one of the best things to come down the pike, when and if one did. There are a couple on the market now, and so we must wonder where things will go from this point on.

One of them seems to be in the area of memory. And, in this instance, "memory" would seem to come in two forms: dynamic memory and disk storage.

Yes, we know that disk storage does not fall into the area of memory *per se*, but it is a viable way to increase the storage capacity of a computer.

We are reminded of an article which Alan J. Morgan wrote in an early issue of *the Rainbow* which spoke about the need for additional memory. Morgan's thesis at the time was that programmers who are able do not need more memory — though they might need more storage.

If you will think about that for a while and add in the fact that most things for which you need memory can be done with a disk as well — sorts, for example — then the line *does* begin to blur between "real" memory and storage "memory."

Couple that to the fact that even though dynamic RAM memory is becoming less and less expensive, "real" memory is still a pretty costly proposition. And that includes bubble memory as well.

Truth to say, although there have been some interesting, from a technical point of view, articles in several journals concerning bubble memory, the fact is that it probably won't be viable on a cost basis for CoCo. Just too expensive to produce on a mass basis.

That leaves our hybrid, disk memory. And, after you hook up four disk drives, what can you do?

Why, do to a hard disk, of course.

We expect that there will be some moves toward a hard disk for CoCo in the coming year. Whether they arise from Tandy Center or elsewhere, it seems certain that this is the way that things will go in the future. And we'll be surprised if *someone* doesn't have a hard disk drive for CoCo available by this time next year.

And what else? Rumors seem very persistant about even newer Color Computers from Radio Shack. We *could* be wrong, but we don't see anything really innovative in the next few months. One of the things Tandy seems to have done is to generate two "lines" of CoCos, with the 64K and the CoCo 2. We think they'll let the dust settle for a little while before things start popping again.

One other trend we see, though, is for more "home" education products to make their appearance. We're not talking about formal educational setting materials, though there will be more and more of those, too. We believe the computer at home is a very viable appliance, if you will, and the development of quality "home education" programs is a very real market, along with the other, more traditional, ones.

* * *

HIGH RESOLUTION screens seem to be all the rage right now. *The Rainbow* has already reviewed at least one of these packages, *Super Screen* by Mark Data Products. Another one, for use with OS-9 from Frank Hogg Labs, is due for review in a future issue.

Now comes still another, called *Hi-Res Screen Pak* from Cer-Comp. The Cer-Comp version allows for all kinds of different displays, including double-wide and double-high characters. It allows use of *PRINT@* and will display up to 255 characters per line. The highest character-per-line displays are not readable, but they do allow you to format page layouts and the like.

* * *

SPEAKING OF GRAPHICS the new idea of the month award goes to All Event Video for its *Comp-U-Trace* product. This is a screen overlay that lets you place a clear, reusable vinyl overlay that you can attach to your screen and then "trace" the same picture with programming. Very interesting.

While on the subject of graphics, there is a new product out from The Micro Works called *Magigraph* for the experienced BASIC and ML programmer. This new program will allow drawing of very detailed graphics, including use of a full set of logical operations and pixel manipulation.

* * *

ANOTHER AREA WHERE we are seeing some rapid growth in the computer world is in the spreading of program-sales stores. Two of the most active firms in this area have been The Program Store and Software City.

What is most interesting to us in this area is that the primary aim of such chains is software, not hardware. In many ways, this breaks the mold of the "traditional" computer store — which pushed one or two brands of computers and, for that reason, carried little more than software for the brands they carried.

Judging from the success of The Program Store and Software City (The Program Store has just opened a franchise in Littleton, Colo. and Software City is moving into London, England), we believe this is a development that will prove interesting in the months to come.

* * *

BRIEFLY NOTED . . .

The Software Connection in Fort Lauderdale has something everyone near a large body of water might be interested in owning: A hurricane tracking program.

Mark Data Products, which pioneered CoCo Adventure games, is out with new one: *Shenanigans,* which is said to combine both text and graphics and which does *not* require a disk system.

Want some customized software? Try bitCards from Chartscan. Their first offering is called *A Christmas Adventure* and it features customized references to the person who receives it.

REDCHASE

By Rich Dersheimer

This program changes the text screen to the orange/red mode, clears the screen to red blocks, and then demonstrates a chaser for the lo-res color graphics.

The color changes can be used to highlight titles and instructions, or if used in groups of eight, for a burst of colors without changing the original block graphics.

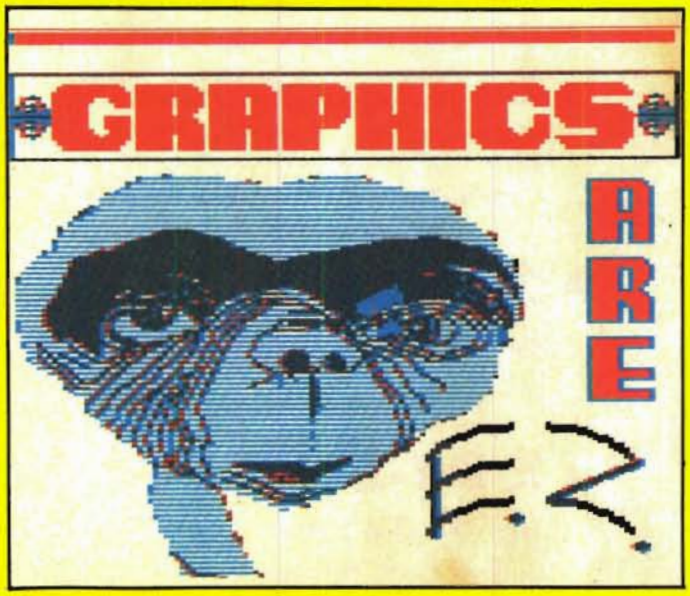
The listing:

```
220.... 018C
400.... 02BF
END... 03ES
```

```
1 ' REDCHASE
2 '
3 ' WRITTEN BY
4 ' RICH DERSHEIMER
5 ' 2203 S.HUGHES
6 ' AMARILLO, TX    79109
7 '
100 CLEAR200,16300
110 'DATA FOR RED SCREEN
120 DATA 86,20,8E,03,FF
130 DATA 30,01,A7,84,8C
140 DATA 05,FF,25,F7,39
150 ' DATA FOR CHASER
160 DATA 8E,04,00,8C,06
170 DATA 00,27,1A,A6,84
180 DATA 81,80,25,08,81
190 DATA F0,25,08,8B,90
200 DATA A7,84,30,01,20
210 DATA E9,8B,10,A7,84
220 DATA 30,01,20,E1,39
230 ' LOAD INTO HIGH MEMORY
240 FOR X=1 TO 50
250 READ A$
260 POKE X+16300,VAL("&H"+A$)
270 NEXT X
280 ' DEFINE USER ROUTINES
290 DEFUSR0=16301
300 DEFUSR1=16316
310 ' TEST THE CODE
320 SCREEN 0,1
330 POKE 359,13
340 A=USR0(0)
350 PRINT@70,"chaser";
360 PRINT@77,"test";
370 PRINT@105,"with";
380 PRINT@110,"red";
390 PRINT@114,"screen";
400 FOR X=0 TO 7
410 N=143+16*X
420 PRINT@172+X,CHR$(N);
430 PRINT@204+X,CHR$(N);
440 PRINT@236+X,CHR$(N);
450 PRINT@268+X,CHR$(N);
460 PRINT@300+X,CHR$(N);
470 NEXT X
480 PRINT@357,"hit";
490 PRINT@361,"spacebar";
500 PRINT@370,"to";
510 PRINT@373,"stop";
520 A=USR1(0)
530 SOUND RND(100),1
540 A$=INKEY$
550 IF A$<>" " THEN 520
560 SCREEN 0,0
570 PRINT@480,"";
580 POKE 359,126
590 END
```

Now, You Can Easily Generate Graphic Data Statements

By Fred B. Scerbo



One of the greatest strengths of the CoCo is the ease with which one can create graphic displays and drawings without having to *POKE* complex geometric formulas to the screen. Commands such as *CIRCLE, LINE, PAINT, DRAW, GET,* and *PUT* can be used to manipulate any sections of the graphic screen so as to display any item which you have the patience to draw. Add to this the release of the X-Pad by Radio Shack and a variety of drawing programs by independent software dealers, and you can create almost any kind of graphic display in up to four *true* colors at one time.

What happens if you use one of these graphic generators to create a detailed display which you wish to incorporate into a BASIC program? Most will allow you to save the contents of the graphic memory to tape (*CSAVEM* "name" 1536,7679,0) or to disk (*SAVEM* "name",3584,9727,0). This can be later loaded back into memory from a program. However, there are several drawbacks to this technique.

First, tape and disk versions can not be interchanged unless they are offset loaded, which can be tricky.

Secondly, you cannot shrink your graphics down to a smaller size.

Thirdly, using the machine language screen dump requires additional I/O from tape or disk.

Fourth, there could be no opportunity for a listing of your graphic display in a magazine such as *the Rainbow*, unless it were a dump of the graphic memory done with *POKE* statements.

Here is where we introduce the *Graphic Screen Data Compiler*. This short program will take whatever graphics you have drawn on the graphic screen and create a BASIC program which will recreate the same graphic from *DATA* statements using the *DRAW* commmand.

Before we go any further we should probably explain exactly how the CoCo's graphics work. When you draw in any *PMODE*, your graphics will remain in the graphics memory until you use the *PCLS* command. You can even wipe your program memory clean with the *NEW* command. Your graphic memory remains untouched.

(You don't believe me? Try this. Write a few lines to create any graphic display with the commands you are familiar with, such as:

```
10 PMODE4,1:PCLS:SCREEN1,1
20 CIRCLE(128,96),20,1
30 GOTO30
```

RUN the program and then enter *NEW* and *LIST*. Now type SCREEN1,1:FOR I=1 TO 2000:NEXT and press [ENTER]. Your graphics are still there.)

Therefore, if we draw a graphic display by whatever means we choose (a program, the X-Pad, a graphics package, etc.), we can load in the *Compiler* without destroying our graphics. We can save only part of the screen if we wish, since the program asks us at which position on the graphic screen we wish to start and end, from vertical position 0 through 192.

While the *Compiler* will work in 16K, it is likely to create a ready to run program which will only run in 32K. (Sorry about that for you 16K folks. Now would be a good time to upgrade.)

Here is how the program works. Each graphic line from 0 to 192 is analyzed pixel by pixel. The *Compiler* then writes two *DRAW* statements counting the number of pixels and their color. Two statements are created per line since a single statement might exceed 255 characters, the maximum length of a *STRING*. When the line has been compiled, the program will write the statement to either tape or disk preceded by a line number and the command *DATA*. The information is written in ASCII form so that when we try to reload the file the *Compiler* creates, we will be loading a ready to run BASIC program. (The CoCo's internal mechanisms recognize this as a BASIC program when it sees the line number.)

(Fred Scerbo is a Special Needs instructor for the North Adams Public Schools. He holds a Master's in education and has published some of the first software available for the Color Computer through his software firm, Illustrated Memory Banks.)

When you run the *Compiler*, you may choose either tape or disk. Simply follow the instructions shown on the screen.

Now let us talk about the drawbacks to this program. The *Compiler* is very slow. After all, it is analyzing a tremendous amount of information and compacting each string as it is created. (The program may take up to half an hour to create the ASCII file so if you are using tape, make sure it is at least a C-60.) Once you have created your file program/BASIC program, you may reload and run it. It is wise to resave the listing since the ASCII version loads more slowly than a normal *CSAVE*.

Notice also that the *Compiler* uses the *PMODE4* and *PMODE3*. If you wish to change to another *PMODE* you must alter the listing yourself.

You will notice that I have included *REM* statements that say speed up and slow down. If you wish, you can use the high speed *POKEs*, especially with disk (while some will). Simply insert *POKE65495,0* to speed up and *POKE65494,0* to slow down. If you use one, however, you must use the other, otherwise you will crash your disk or mess up your tape. If you are really daring, use *POKE65497,0* to speed up and *POKE65496,0* to slow down. This will disconnect your screen and fill it with garbage while the data is being analyzed so do not press break or you will lose the program.

You will gain some speed with these *POKEs* but you are probably better off being patient. Walk away from your CoCo and let it do the work for you. After all, that is why you bought a computer. Just imagine how long it would take to write these *DRAW* statements by hand!

Earlier I mentioned using this program to shrink your graphics. Here's how.

Take the BASIC program created by the *Compiler* and use the following lines instead:

```
10 PMODE1,1:PCLS:SCREEN1,1
40 DRAW"S2BM0,"+I$+A$:DRAW B$
55 PMODE4,1:SCREEN1,1
```

Keep the rest of the lines the same. What you have done is draw only halfway across the screen since you have changed size from 4 to 2 in the *DRAW* statement. By switching from *PMODE1* to *PMODE4*, you will have created a graphic half the size of the original. You can rerun this graphic through the *Compiler* and create the lines needed to recreate this smaller graphic inside another BASIC program. Experiment.

What uses can this be put to? You can generate small parts of the graphic screen which can be merged with other BASIC programs. Create a complex small graphic which you can read into a *GET* and *PUT* statement in your own games. With a little manipulation, you can generate *DRAW* statements small enough to fit into a 16K game. The process is slow, but once it is done, the results can be pleasing and you can use a graphics package to create detailed, arcade-like graphics for your own BASIC programs.

In time, I hope some of you will use this *Compiler* to submit some complex graphic computer art for Lonnie to run on his Radio Shack Color Graphic Paint printer for inclusion in *the Rainbow*. This could even become a monthly competition or listing. Then all of you can really show off your CoCo's power to those friends of yours who bought a fruity computer. (Just dare them to try it on their machine.)

Included with this article is a printout from *the Rainbow's* CGP-220 Ink-Jet Printer created from a very long but detailed graphic listing which displays everyone's favorite alien in *PMODE4*. (Remember this is just an artistic display and cannot be used for promotional purposes as the actual image is protected from such unauthorized use for profit. Still, cartoonists use this image for artistic expression, so that is what we are doing here. This is for your own use only.)

The listing for *DRAW-EZ* is not reproduced in the magazine, due to its extreme length, but *is* included on November's *Rainbow On Tape*. *DRAW-EZ* requires 32K since there is so much detail in the drawing. Those of you with 16K might try part of the listing just to see the results partly displayed.

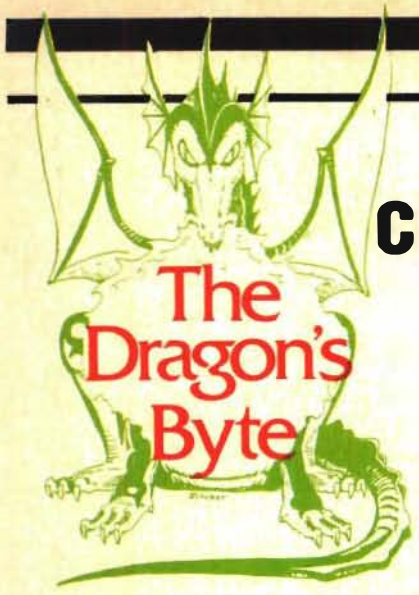
Now let's see what all of you can come up with. Let's see some real computer art submitted to *the Rainbow* for future issues. You have the tools to do it now.

The listing:

```
130 ... 0105
300 ... 0367
570 ... 0577
END .. 077C
```

```
10 CLEAR1000
20 CLS
30 PRINTSTRING$(32,42);
40 PRINT"  GRAPHIC SCREEN DATA C
OMPILER"
50 PRINTTAB(8)"BY FRED B.SCERBO"
60 PRINTSTRING$(32,42);
70 PRINT" ENTER PROGRAM NAME TO
BE SAVED"
80 PRINT" UP TO 8 LETTERS: "
90 PRINTSTRING$(32,42);
100 PRINT@178,"";
110 LINEINPUT PN$
120 IF LEN(PN$)>8 THEN RUN
130 IF LEN(PN$)=0 THEN RUN
```

```
140 PRINT@228,"(T)APE OR (D)ISK
OUTPUT?"
150 PRINTSTRING$(32,42);
160 IF INKEY$="T"THEN170ELSEIF I
NKEY$="D"THEN190ELSE160
170 W=-1:PRINT"  PRESS <ENTER> W
HEN RECORDER":PRINTTAB(11)"IS RE
ADY":PRINTSTRING$(32,"*");
180 IF INKEY$=CHR$(13)THEN200ELS
E180
190 W=1:PN$=PN$+"/BAS"
200 OPEN"O",#W,PN$
210 FORI=1TO100:NEXT
220 PRINT#W,"5 CLEAR500"
230 PRINT#W,"10 PMODE4,1:PCLS1:S
CREEN1,1:PMODE3,1"
240 CLS:INPUT"STARTING LINE ( 0
- 192 ):";SP:
250 IF SP<0 THEN240 ELSEIF SP>19
2 THEN240
260 PRINT"ENDING LINE ("SP" - 19
2 ):";:INPUT EP
270 IF EP<SP THEN 240 ELSEIF EP>
192 THEN260
280 A$="20 FORI="+STR$(SP)+"TO"+
STR$(EP):PRINT #W,A$
290 A$="30 I$=STR$(I):READ A$:RE
AD B$":PRINT #W,A$
300 A$="40 DRAW"+CHR$(34)+"S4BM0
,"+CHR$(34)+"+I$+A$:DRAW B$":PRI
NT #W,A$
310 A$="50 NEXT I":PRINT #W,A$
320 A$="60 GOTO60":PRINT #W,A$
330 PP=1000
340 PMODE3,1:SCREEN1,1
350 FORY= SP TO EP
360 REM SPEED UP
370 FOR L=0TO128STEP128
380 CLS
390 A$=""
400 T$=STR$(PP)+"DATA"
410 C=PPOINT(0,Y):T$=T$+"C"+STR$
(C)+"R"
420 PRINT"NOW DECODING LOCATION
# ";Y
430 FORI=0+L TO 128+L
440 PRINT@63,T$;
450 D=PPOINT(I,Y):IF D=C THEN490
460 H$=STR$(V)+"C"+STR$(D)+"R":P
RINTH$;:GOSUB670
470 T$=T$+H$:H$=""
480 C=D:V=1:GOTO500
490 V=V+1
500 NEXTI
510 H$=STR$(V):GOSUB670:T$=T$+H$
520 PRINT@64,T$:V=0:A$=T$:T$=""
530 T$=A$
540 K=LEN(T$)
550 FORYY=1TOK:R$=MID$(T$,YY,1)
560 IF R$=" "THEN580
570 P$=P$+R$:PRINT@64,P$;
580 NEXT YY:A$=P$:PRINT@64,P$;"
  ";:P$="":PP=PP+10
590 REM SLOW DOWN
600 PRINT #W,A$
610 REM SPEED UP
620 NEXTL
630 NEXTY
640 REM SLOW DOWN
650 CLOSE #W
660 GOTO710
670 TR=LEN(H$):FOR KL=1TO TR:JJ$
=MID$(H$,KL,1):IF JJ$=" "THEN690
680 WR$=WR$+JJ$
690 NEXT KL
700 H$=WR$:WR$="":RETURN
710 CLS:PRINT@160,STRING$(32,42)
;
720 PRINT"YOU BASIC GRAPHIC PROG
RAM IS NOWSAVED AND READY TO BE
RELOADED  INTO YOUR COMPUTER'S M
EMORY."
730 PRINTSTRING$(32,42);:PRINT:P
RINT:PRINT:PRINT
740 END
750 REM ****GRAPHIC SCREEN DATA
COMPILER BY FRED B. SCERBO, USED
BY THE PERMISSION OF ILLUSTRATED
MEMORY BANKS, COPYRIGHT (C) 1983
```



Computer Supported Combat Is No Fantasy

By Bill Nolan
Rainbow Contributing Editor

Welcome back to the land of fantasy gaming. You may recall that we have been putting various portions of the tedious work associated with record-keeping and character creation process on the computer. The built-in problem with this has been simply that the existing fantasy role-playing games were never designed or written with computers in mind.

Having realized that, we have been working on a combat system designed from the ground up to be computer supported. That has been very well received, so what you see here is the start of a grander (perhaps grandiose) plan for a whole fantasy gaming *system* designed from the ground up to be computer supported.

This summary of the new game is not cast in concrete, and some of what is below will possibly be altered in the future, but we have been test playing it for about two months in our own group, so you will find most of it to be workable as is.

There are eight character attributes, and each of these are found by rolling five four-sided dice, throwing out the lowest two, and adding up the highest three. In the listing below you will find a neat little program to roll up characters and also check to see which characters qualify to be which races and classes.

You will notice that when a character is rolled up, you have a choice to reroll or continue. If you choose to continue, you will find the program ends with an error. That's because I haven't written the rest of the program yet. Eventually the program will do a *very* complete character generation. Well then, on to the summary.

CHARACTER ABILITY SCORES

Physical Beauty	Agility
Intelligence	Manipulation
Faith	Stamina
Speed	Strength

CHARACTER CLASSES

Warrior	Arcanist (white, black, and green)
Thief	Guardian (white, black, and green)
Rogue	Champion (white, black, and green)

(Bill Nolan teaches computer science at a local college, moderates a fantasy game once a week, and is a principal in Prickly-Pear Software)

THE MAGIC SYSTEM—A Quick Overview

The colors of magic are complex, and we will go into them in detail next time. To quickly summarize, players must align themselves with the white, green or black if they are an Arcanist, Guardian, or Champion. The main thing to keep in mind here is that these colors are not related to the concept of alignment used in many games. A character's choice of magic color is unrelated to good or evil. There are several other colors of magic known, but they may not be used by player characters.

There are three levels of magic. These are trivial or trick spells, battle spells, and regular spells. There are also several levels of power among regular spells. A spell user can cast spells each day up to the limit of his force points. These are figured as follows: Faith + Intelligence + Stamina + (Level * 3). Trivial spells use one point. Battle magic uses two points, and regular spells use their power level times five points. If the caster is a warrior, rogue, or thief, then double force points are used up in casting spells.

Magic is written in a special language that can be learned by anyone having an intelligence of nine or more. Once the language is learned, any character may try to cast spells as detailed below. If you cast spells past the limit of your force points, the extra is deducted from your hit points, and these will be recovered at the rate of one per week. There has never been a case recorded of them being recovered more quickly.

Arcanists and Guardians can form a chain of power. This will combine their force points and chance of success.

DESCRIPTIONS OF CHARACTER CLASSES

Warrior: To be a warrior, a character must have agility of seven, stamina of nine, and strength of nine. A warrior has a combat adjustment of six. If a warrior learns the magic language, he can try to cast spells of trivial and battle magic.

Thief: To qualify as a thief, the character must have an intelligence of eight, speed of eight, agility of nine, and manipulation of ten. The combat adjustment is three.

Thieves have the following abilities: PICK LOCKS 30% + (5% * (level -1)) + (5% if manipulation is 11 or 15% if 12). PICK POCKETS 30% + (5% * (level -1)) + (5% if manipulation is 11 or 15 % if manipulation is 12). STALKING 40% + (5% * (level -1)) + (5% if intelligence is 10 or more) + (5% per point above 10 of agility). HIDING 30% + (5% * (level -1))

+ (5% if intelligence is 10 or more) + (5% per point of agility above 10).

When a characteristic reaches 85%, it begins going up 2% per level, and can never exceed 95%. If you fail to pick a pocket, the intended victim always notices the attempt. Also, picking pockets is just that. It confers no ability for more complex maneuvers. It is impossible to stalk or hide in noisy or shiny armor. If a thief learns the magic language, he can try to cast spells of trivial and battle magic.

Rogue: To be a rogue requires physical beauty of ten, intelligence of ten, speed of eight, agility of nine, manipulation of ten, stamina of nine, and strength of nine. This character has the combat adjustment of a fighter and the thieving skills of a thief. Rogues can cast trivial and battle magic spells, as well as regular spells up to first power level. Check out the points for rising in levels before choosing this class. No complaints later!!!

Arcanist: If you have intelligence of ten, manipulation of nine, stamina of eight, and faith of nine, you can be an arcanist. Combat adjustment is zero. Arcanists must choose between the green, black, and white, and can never turn back from a chosen way. The chance of successfully casting a spell (assuming directions are at hand) is: Intelligence * 5 + manipulation * 3 + Faith + (level -1) * 10. This is for a spell of first level of power. The chance goes down by 25% for each level higher. However, there is always a 3% chance of failure or success. Each time you correctly cast a spell, your chance on that spell goes up 2%. If you miss a spell by 25% or more, you must roll on the dreaded spell mis-fire table to see what happened.

Guardian: You need an intelligence of nine and a faith of 12 to be a guardian. A guardian must choose between the green, black, and white, and can never turn back from the chosen way. Combat bonus is two. A Guardian can turn aside undead if he successfully rolls a saving throw. However, the level of the undead is added to his roll. The chance of successfully casting a spell (assuming directions are at hand) is: Intelligence * 3 + Faith * 5 + (level -1) * 10. This is for a spell of first level of power. The chance goes down by 25% for each level higher. However, there is always a 3% chance of failure or success. Each time you correctly cast a spell, your chance on that spell goes up 2%. If you miss a spell by 25% or more, you must roll on the dreaded spell mis-fire table to see what happened.

Champion: These fighters for the cause need an intelligence of ten, faith of 11, speed of ten, agility of ten, stamina of ten, and strength of ten. No, we won't see many of these! These fighters have a combat adjustment of 6, and because of the force being with them, they are +1 to hit. They can cast regular spells of the arcanist up to first power level, and guardian spells up to second level. However, only spells of the proper color can be cast. Champions are immune to poison of any kind. If they leave the way of their color, they will invariably sicken and die of unknown causes.

For All Classes: Hit points are Faith, Stamina, and Strength added together. When hit point level reaches zero, the character is unconscious. Death occurs at double the hit point level below zero. Each level after the first adds one hit point. Females subtract two from strength and add two to manipulation. They get the benefit both ways. If the deduction drops them below needed strength, they qualify anyway, and if the manipulation brings them up to qualifying level, that's fine too.

Climbing: All classes climb the same. For a rope or easy to climb wall, the formula is (Agility + Strength + Level) * 3.

For easier or harder conditions, the adjustment will be provided by the moderator. Use percentile dice.

Finding Secret Doors: :All races have the same percentage of chance. If not actively looking, the chance is (Intelligence + Level) * 2. If looking, the chance doubles.

Fin and Remove Traps: Everyone has the same chance. (Intelligence + Manipulation + Level) * 2. Roll twice — once to find and again to remove. The moderator rolls the chance to find. Use percentile dice.

Races: There are three races which can be player characters. These are Human, Elf, and Dwarf. Anyone can be human — there are no qualifying scores.

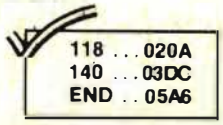
Elf: These need physical beauty of 11, intelligence of ten, agility of ten, and manipulation of ten. They subtract one from the initial rolls of the dice for strength and stamina, and the result must be less than nine or it will be reduced to make this so. These adjustments are made after class qualifying, and the hit points are reduced when applicable. Elves have *wood sense*, which operates only above ground. They are never lost, they can find any available food or water, and they sense large (man-size or bigger) animals within 150 feet. It is not possible to sneak up on an awake elf above ground.

Dwarf: If you covet Dwarfhood, you need a strength of eight. Dwarves subtract one from speed and physical beauty, and if either of these ends up higher than eight, it will be reduced to that level. Again, this is done after class qualifying. Dwarves have *stone sense* when below ground. They can sense pits in the floor large enough to fall into, and hollows in the rock larger than ten feet across within 50 feet — even if blocked by rock. They never bump into walls underground, even in pitch blackness. They can't see them, but they know they are there. They are never lost underground, and always know the distance to the surface.

In the case of both elves and dwarves, the "never lost" assumes that they went to a place while aware. If they are teleported or carried unconsious, they won't know where they are.

Saving Throws: If a saving throw is required, the character must roll a 30-sided die and obtain a number lower than their current force point level divided by two and rounded down. A 30 always misses and a one always succeeds. The moderator will specify plusses or minuses as necessary.

That's about it for this month. Below is the character generating program. Remember, it works as is, but isn't finished yet. We will be adding on to it next month. In the meantime, I want to warn you to beware of the dragons in this game. They are *very, very* powerful. No more easy Dragon treasure! By the way, the tentative name for the new game is *Dragon Magic,* suggested by Pat Zircher, the artist who drew the pictures at the head of this column every month. If you have a better name, let me know.



The listing:

```
1 '********DRAGON MAGIC********
2 '******CHARACTER CREATOR******
3 '*******COPYRIGHT 1983********
4 '****PRICKLY-PEAR SOFTWARE****
5 '*****ALL RIGHTS RESERVED*****
6 X=RND(-TIMER)
10 DIM C(8),C$(8)
20 FOR X=1 TO 8:READ C$(X):NEXT
100 CLS:PRINT" PRESS ANY KEY TO
ROLL":K$=INKEY$:GOSUB 9000:PRINT
" COMPUTING ROLL";:GOSUB 199
110 CLS:FOR X=1 TO 8:PRINTC$(X),
C(X):NEXT X
112 HP=C(3)+C(7)+C(8):PRINT"HIT
POINTS",HP:CC=0
115 IF C(1)>9 AND C(2)>9 AND C(4
)>7 AND C(5)>8 AND C(6)>9 AND C(
7)>8 AND C(8)>8 THEN PRINT" ROGU
E",:CC=1
118 IF C(5)>6 AND C(7)>8 AND C(8
)>8 THEN PRINT" WARRIOR",:CC=1
120 IF C(2)>7 AND C(4)>7 AND C(5
)>8 AND C(6)>9 THEN PRINT" THIEF
",:CC=1
122 IF C(2)>9 AND C(3)>8 AND C(6
)>8 AND C(7)>7 THEN PRINT" ARCAN
IST",:CC=1
124 IF C(2)>8 AND C(3)>11 THEN P
RINT" GUARDIAN",:CC=1
126 IF C(2)>9 AND C(3)>10 AND C(
4)>9 AND C(5)>9 AND C(7)>9 AND C
(8)>9 THEN PRINT" CHAMPION",:CC=
1
127 IF CC=0 THEN PRINT" COMMONER
",
128 IF C(1)>10 AND C(2)>9 AND C(
5)>9 AND C(6)>9 THEN PRINT" ELF"
,
129 IF C(8)>7 THEN PRINT" DWARF"
,
130 PRINT:PRINT"PRESS 'R' TO RER
OLL OR 'G'":PRINT"TO GO ON.";:K$
=INKEY$
140 K$=INKEY$:IF K$<>"R" AND K$<
>"G" THEN 140 ELSE SOUND 150,1:I
F K$="R" THEN 100
150 CLS:PRINT"THIS PART ISN'T DO
NE":END
199 FORC=1TO8:C(C)=0:FORC=1TO8:D
1=RND(4):D2=RND(4):D3=RND(4):D4=
RND(4):D5=RND(4)
200 HH$(1)=STR$(D1):HH$(2)=STR$(
D2):HH$(3)=STR$(D3):HH$(4)=STR$(
D4):HH$(5)=STR$(D5)
210 FOR X=1 TO 4:IF HH$(X)>HH$(X
+1) THEN 250
220 NEXT X
230 C(C)=(VAL(HH$(3))+VAL(HH$(4)
)+VAL(HH$(5)))
240 PRINT".";:NEXT C:RETURN
250 X$=HH$(X):HH$(X)=HH$(X+1):HH
$(X+1)=X$:GOTO210
9000 IF INKEY$="" THEN X=RND(0):
GOTO 9000:ELSE SOUND 150,1:RETUR
N
10000 DATA PHYSICAL BEAUTY,INTEL
LIGENCE,FAITH,SPEED,AGILITY,MANI
PULATION,STAMINA,STRENGTH
```

An Introduction To Synchronous Data Transmission

By Harry Hardy

In the April and June issues we took a look at asynchronous transmission, in particular as to its relationship to the Color Computer. As we now know, asynchronous transmission can be implemented rather inexpensively and is rather efficient, at least for our needs. We also know that the error detection methods are not foolproof and error recovery is seldom implemented. Finally, asynchronous transmission is usually limited to speeds less than 2400 bits per second (BPS). Probably 300 and 1200 BPS are more commonly used. I have heard that faster speeds can be used, but I'm not familiar with those devices. Speeds up to 9600 BPS are used but the only asynchronous terminals I've seen operating at speeds this high are connected directly to a computer for in-house use only or they have some kind of protocol converter to convert from asynchronous to synchronous transmission. We'll look at that later. Well, that's enough about asynchronous terminals for now, let's look at some newer technology.

Since the theme this month is communications I thought I would introduce you to another type of data transmission where speeds up to 56000 BPS (56K BPS) are not uncommon, and even speeds up to 1.54 million BPS and faster may be found. This type of transmission is called *synchronous* transmission.

Before we look at this technology, I should let you know that my background is with large IBM mainframes using either their terminals or what are known as plug-compatible terminals, mainly those made by Teletype Corporation. Since this environment (IBM) is my forte, the examples we will be looking at will be of those device types in particular, however the principles involved apply to all devices capable of synchronous transmission, also some of the examples will apply to asynchronous transmission; where this overlap occurs I will point them out.

A good place to start is to look at how synchronous transmission differs from asynchronous. You may recall that I defined asynchronous transmission as one that lacked any continuity between transmitting and receiving stations, and in order to achieve this continuity or synchronization *start* and *stop* bits were used so the receiving station could get synchronized on the character being transmitted so it would know when to start assembling bits into a character. Let's briefly look at how efficiency is affected in an asynchronous environment.

(Harry Hardy, who has 15 years experience as an applications and systems programmer in telecommunications is currently employed as a telecommunications consultant by a large company that provides data communications products and services.)

Assume we are using an 8-bit character which will be preceded by a start bit and end with a stop bit. That gives us a total of 10 bits per character. Since each and every character will require these framing bits, we have a 20 percent overhead on each character being transmitted. As you have probably figured out, if you are transmitting a lot of data, this 20 percent overhead can really delay the actual processing of this data by your program. For example, if your program requires 100 characters of information to process a given transaction, 200 of the 1,000 bits, or 25 characters, are overhead to your program to get the 100 characters you require.

How does synchronous transmission differ from this, you ask? Well, to help understand this let's look at the RS-232C interface you have heard about. This interface is a standard that has been agreed upon by the Electronics Industries Association (EIA) for interfacing different manufacturers' terminal and computer equipment. The RS-232C port on your Color Computer meets this standard, even though in just about the most simple form possible to perform data communications.

Figure 1

PIN	FUNCTION	CIRCUIT ID
1	GROUND	AA
2	TRANSMITTED DATA	BA
3	RECEIVED DATA	BB
4	REQUEST TO SEND	CA
5	CLEAR TO SEND	CB
6	DATA SET READY	CC
7	SIGNAL GROUND	AB
8	DATA CARRIER DETECT	CF
9	POSITIVE DC TEST VOLTAGE	
10	NEGATIVE DC TEST VOLTAGE	
12	SECONDARY CARRIER DETECT	SCF
13	SECONDARY CLEAR TO SEND	SCB
14	SECONDARY TRANSMITTED DATA	SBA
15	TRANSMITTER CLOCK	DB
16	SECONDARY RECEIVED DATA	SBB
17	RECEIVER CLOCK	DD
19	SECONDARY REQUEST TO SEND	SCA
20	DATA TERMINAL READY	CD
21	SIGNAL QUALITY DETECT	CG
22	RING INDICATOR	CE
23	DATA RATE SELECTOR	CH
24	EXTERNAL TRANSMITTER CLOCK	DA
25	BUSY	

RAINBOWfest

FREE RAINBOW POSTER For First 200 Tickets Sold.

The fun and excitement of *RAINBOWfest* is coming your way . . . and now there will be a *RAINBOWfest* near you!

For the 1983-84 season, we've scheduled four *RAINBOWfests* in four parts of the country. Each one will offer fun, excitement, new products, seminars and information for your CoCo! And for those who (perish the thought) don't like CoCo as much as you, we've scheduled each *RAINBOWfest* in an area that will provide fun and enjoyment for the whole family.

Just look at this great lineup:

Fort Worth, Texas — October 14—16. We'll kick off the *RAINBOWfest* "season" with a visit to CoCo's home — Radio Shack's headquarters in Fort Worth. Our CoCo Community Breakfast will feature Ed Juge, Radio Shack's director of computer merchandising, as the guest speaker. And, we hope to be able to offer tours of Tandy Center for those who will be at what promises to be an exceptional *RAINBOWfest* kickoff.

Long Beach, California — February 17-19. What a way to get away from the winter doldrums! And what better place than sunny Southern California with thousands of fellow CoCo owners! Fly with CoCo to the sun for the winter. Top flight seminars will draw on many local CoCo experts.

Cherry Hill, New Jersey — April 6—8. Just 6.3 miles from the Liberty Bell, *RAINBOWfest* comes to the populous northeast. It's a close drive from New York, Boston, Philadelphia, Washington and Baltimore.

Chicago — June 22—24. We'll play *RAINBOWfest* again, CoCo! This is the site of CoCo's *very first show* this spring. And right next to the world's largest indoor shopping mall.

Every show will be held at a Hyatt-Regency Hotel and all will be offering special rates for *RAINBOWfest*. Every show will open at 7—10 p.m. Friday, run 10 a.m.—6 p.m. Saturday and close with an 11 a.m.—4 p.m. session Sunday. Each will have a CoCo Community Breakfast featuring an outstanding national speaker from the Color Computer World. And each exhibition will be interspaced with a number of seminar sessions on all aspects of CoCo —from writing in machine language to making your BASIC work better.

But most of all, there will be exhibitors. Lots of them. All ready to demonstrate products of every kind. Some with special programs and hardware items to introduce. Others with show specials.

Tickets can be secured directly from *the Rainbow*. We'll also send you a special reservation form so you can get your special room rate.

Come to *RAINBOWfest* . . . help us all celebrate CoCo Community at its finest!

RAINBOWfest—Fort Worth
DATES: Oct. 14—16, 1983
HOTEL: Hyatt Regency, Ft.Worth
ROOMS: $55 per night single/double
KEYNOTE: Mark Yamagata
Director of Computer Merchandising
Personal Computer Products
Radio Shack
Advance Ticket Deadline: Oct. 10, 1983

RAINBOWfest—Long Beach
DATES: Feb. 17—19, 1984
HOTEL: Hyatt Regency, Long Beach
ROOMS: $59 per night single/double
KEYNOTE: Bob Albrecht
Advance Ticket Deadline: Feb. 13, 1984

RAINBOWfest—Cherry Hill, NJ
DATES: April 6—8, 1984
HOTEL: Regency-Hyatt Cherry Hill
ROOMS: To Be Announced
KEYNOTE: To Be Announced
Advance Ticket Deadline: Mar. 30, 1984

RAINBOWfest—Chicago
DATES: June 22—24, 1984
HOTEL: Regency-Hyatt Woodfield
ROOMS: To Be Announced
KEYNOTE: To Be Announced
Advance Ticket Deadline: June 18, 1984

RAINBOWfest

Make checks payable to:
the Rainbow

MAIL TO:

RAINBOWfest
P.O. BOX 209
Prospect, KY 40059

YES, I'm coming to *RAINBOWfest* in

☐—Ft. Worth ☐—Long Beach ☐—Cherry Hill ☐—Chicago.

Please send me:

_____ three day tickets at $8 each total _____
_____ one day tickets at $6 each total _____
_____ breakfast tickets at $11 each total _____

Handling Charge $1.00

TOTAL ENCLOSED (U.S. FUNDS ONLY, PLEASE) $ _____

—Also send me a hotel reservation card for:
☐—Ft Worth ☐—Long Beach ☐—Philadelphia ☐—Chicago

NAME *(please print)* _____
STREET & NUMBER _____
CITY & STATE _____
TELEPHONE _____ ZIP CODE _____
COMPANY _____

Orders sent less than two weeks prior to show opening will be held for you at the door.

VISA, MasterCharge, American Express accepted.

My Account # _____ Ex. Date _____

Signature _____

RAINBOWfest Fort Worth

Seminar Program And Speakers

Don Inman — **Using And Teaching LOGO**

Don is one of the most respected names in the Color Computer field and an expert on graphic techniques in both BASIC and assembly language. His Using Graphics *column appears monthly in* the Rainbow.

Tim Purves — **Using Voice And Sound Synthesis**

Tim, who writes for Computer Shack, is the author of Fury *and* CIII.

Steve Blyn — **Programming For Education**

Award-Winning Rainbow *columnist and teacher, Steve has written dozens of educational programs for Computer Island.*

Frank Hogg — **Advanced Operating Systems**

Frank is a pioneer in OS-9 and FLEX systems and is president of Frank Hogg Laboratory.

John Grabbard — **Writing Adventure Games**

John, who writes programs for Spectral Associates, is the author of Keys To The Wizard *and* Android Attack.

Andy Larson — **Education and Software Development**

Andy, of Follett Library Book Company, has long been active in developing software for educational uses.

Bill Nolan — **Fantasy Gaming And CoCo**

DungeonMaster, programmer and Rainbow *Columnist, Bill has developed a complete* Dungeons and Dragons *program for CoCo.*

Tom Nelson — **Start Your Own Software Business**

Tom is a Rainbow *columnist and one of the owners of Nelson Software Systems.*

Frank Brandon — **CoCo Business Applications**

Frank, who will be joined in this seminar by business programmer Dennis Mitchell, is principal of Color Software Services, *a division of* Brantex, Inc..

PLUS...RAINBOWfest's "CoCo Community" Breakfast featuring Mark Yamagata, Radio Shack's New Director Of Computer Merchandising for Personal Computer Products.

Mr. Yamagata has just been appointed to this position, which among other responsibilities, has overall supervision for the Color Computer. This will be among his first public appearances in his new capacity.

Additional seminars are planned as well. Admission to all seminars is at no charge. See registration form for admission prices to exhibit area and breakfast.

COME TO RAINBOWfest FORT WORTH —SEE COCO'S OWN HOMETOWN!

If you look at the cable you connect to your computer and your modem, you can see the obvious differences in plugs. The computer end is round and has only four pins, while the other end is flat and rectangular in shape. Although there may be only four pins in that end, there is space for 25 of these pins. Figure 1 lists these pin assignments by number and EIA circuit identification. Note that all pins are not assigned; however, some modem manufacturers do make use of these unassigned pins. We will not discuss those pin functions here since assignment purposes may differ by modem manufacturer.

Your Color Computer uses pins 2, 3, 7 and 8, a very simple EIA interface for data communications. Synchronous and some asynchronous transmission devices use several of the other pins. Let's see which ones as we proceed.

Synchronous transmission, in addition to the transmit data lead (pin 2), requires either a separate clock lead from the transmitting station, or a modem that includes the clocking information when it encodes the data, converts from digital to analog. In other words it is up to either the DTE or the modem to provide the timing, or clock, as it may be referred to, that was accomplished by using start/stop bits in asynchronous communications. At the transmitting end this clock signal, if provided by the computer, is received on pin 24 from the computer. This signal is then put on pin 2 at a different frequency level than the data. If the modem is providing the clocking, then the clock signal is received from circuit DB (figure 1) or pin 15 on the modem, not the computer. This clock signal is then placed on pin 2 just as before. At the receive end, this signal, data and clock, are found on pin 3. Let's not get into the electronics of how this is done, but what happens is that, at the receive end, this timing, or clocking, is recovered from the sidebands of the received signal to indicate to the DTE the appropriate instant to sample the data stream on the received data lead, pin 3, for a character or information. This recovered clock signal is then placed on pin 17 for the receiving station. This inclusion of the clock "beside" the data stream provides the synchronous continuity that is lacking in asynchronous communications. By providing this continuity we have defined the term "synchronous transmission." By the way, in probably 99 percent of all installations the modem provides this clocking, not the DTE.

By eliminating the start/stop bits, we have created a couple of side effects.

One is good. While the other may seem bad, it is easily overcome. First the good; since all bits are used to represent data, we have eliminated the 20 percent overhead or the 20 percent waste characteristic of asynchronous communications. Now the other side effect. As you recall I said that those start/stop bits served another purpose besides timing, that being framing bits. This meant that the presence of a start bit indicated to the receiving station to assemble the next eight bits into a character, the following stop bit indicated to start monitoring for the next start bit. Since these framing bits are absent in synchronous communications we must provide another method of determining which group of bits constitutes a character. Sound like a problem? Not really—in fact, the solution is quite simple. Let's see how this is accomplished.

The framing of each character is accomplished by defining a synchronization character, commonly called a "syn" character. This syn character is usually one that has a bit pattern different from that of any of the regular characters that are to be transmitted in a data stream. This is done for reasons that will become obvious. We wouldn't want the DTE to think that the bit pattern for the letter A is a syn character. This would create a lot of problems, as you can see. Each reception of the letter A would indicate to the receiver to start assembling bits that follow as a character and to ignore the letter A. If your DTE transmits in ASCII, the syn character is made up of the bits 01101000. In the world of IBM, this character would be 00110010. Why the difference? IBM equipment uses a code called EBCDIC, extended binary coded, decimal interchange code. This code uses all eight bits for data, therefore 255 possible characters can be represented where only 127 characters are available with ASCII. Although their synchronous devices talk EBCDIC, their computers can support certain synchronous devices using ASCII.

Back to our syn character: No matter which one we use it will precede the data characters in all transmissions. It seems we've reduced our overhead quite a bit by sending a syn character before each block of data. Well, there's a little more to it. Suppose the first syn character loses a bit along the way. We then lose our data, for the DTE wouldn't know it was to collect the following bits as data. Well, to allow for this condition we send one more syn character than is required. Typically one syn character is required, therefore our synchronous device will send two syn characters. I know, you don't have to remind me—what if both syn characters are destroyed? Well, the line has to be drawn somewhere. We could send 25 syn characters, but then we would be approaching our 20 percent overhead again, so what's the point? Just like in asynchronous transmission, why not send 20 stop bits in case the first 19 are destroyed. Ridiculous, huh?

So, we will send two syn characters followed by our data, correct? Well, let's look a little deeper and see what we can find (sounds like we may be losing ground on our savings). Actually each synchronous transmission has four characters of overhead. Besides the two syn characters, we have a leading and a trailing PAD character. The leading pad bits are 01010101 and the trailing pad bits are all ones, to the computer. From the computer, all transmissions must have at least two syn characters, data, and a trailing PAD character. This PAD has the format of 1111xxxx where x is either a 1 or 0. So from the computer we have three characters of overhead. In addition to this, two syn characters are automatically sent every second during transmission to or from the computer. This is done to make sure the model clocks are still in sync and the data is being sampled at the correct interval.

Does it sound like we haven't gained any in the overhead department? Let's look at a situation and see if that is really the case. Consider the speeds at which we transmit in a synchronous environment. For a computer to terminal data link, the speeds are usually 2,400, 4,800 and 9,600 BPS—7,200 BPS may also be found. Probably 4,800 BPS is the most common speed with 9,600 rapidly gaining in popularity.

Let's say we are running at 4,800 BPS and we have 594 characters of data to transmit. We also have four overhead characters, those being two PADS and two syn characters, and possibly two syn characters will be sent in the middle of our data. Add all those up and you can see we have 600 characters with eight bits each for a total of 4,800 bits to transmit. Of these 4,800 only 48 are overhead. That works out nicely, for depending on the distance between sending and receiving devices, we can send this data in one second.

Now, take those same 594 8-bit characters and send them asynchronously. We would still have 4,752 bits like before, however we now have an additional two bits per character, or 1,188 bits overhead for a total of 5,940 bits. Besides this excess overhead we see that we can't even send the data in one second, assuming we could transmit asynchronously at 4,800 BPS. For you mathematicians, that's one percent versus 20 percent of overhead. Let's not get too anxious; in short blocks of data we would still have those 32 bits of overhead. In those instances—I didn't figure one out—the overhead in synchronous transmission would outweigh the asynchronous overhead; however, in a busy interactive system that isn't likely to happen very often during the actual transfer of data between computer and terminal.

Now that we have some idea of how synchronous transmission works, what about the line discipline or protocol that is used to govern this type of transmission? There are different types of synchronous protocols, but I will focus on only one of them here. The reason is that, although the Color Computer doesn't support synchronous communications, other personal computers do. Apple, Radio Shack's Models 12 and 16, and IBM's PC offer a package that supports binary synchronous protocol. Maybe you own one of those computers in addition to the Color Computer or maybe you have been thinking about getting one. In either case, let's look at some of the concepts of this protocol.

Binary synchronous, or bisync as it is more commonly called, has an extensive set of control codes to ensure that the transfer of data is done in a very accurate and precise manner. In addition to a choice of error detection schemes, it also employs a complete error recovery procedure. There are three types of bisync configurations, these being: point-to-point private line, point-to-point switched line, and a private line multipoint configuration.

The choice of error detection methods is dependent on the type of terminal. If the terminal uses ASCII to represent data, then VRC and LRC are used, resulting in a one-byte block check character. If the terminal uses EBCDIC then the CRC-16 method of error detection is used resulting in a two-byte block check character (BCC). CRC or Cyclic Redundancy Check is an error detection method in which the check character is generated by taking the remainder after dividing all the serialized data bits of a block by a predetermined number. With CRC-16, the 16 tells us how many bits are in the BCC, providing error detection of bursts up to 16 bits in length, with more than 99 percent of error bursts greater than 16 bits being detected. Therefore, systems that use the CRC method of error detection are more effective at detecting errors than those that use VRC/LRC.

As I said, there are several control codes used by bisync to control data flow. Let's define a few of them.

SOH — Start of heading.
STX — Start of text.
ETX — End of text, terminates a block of data and indicates BCC character(s) is next.
EOT — End of transmission, indicates the end of a message, is also used to respond "nothing to transmit" to polling sequence. More on that later.
NAK — Negative acknowledgement indicates previous block received with an error, retransmit data.
ACK0/ACK1 — Positive acknowledgement indicates previous transmission was received error free. ACK0 is used to acknowledge even-numbered blocks, ACK1 is used for odd-numbered blocks.
WACK — Wait before transmit positive acknowledgement indicates that previous block is error free, but receiver is not ready to receive next block.
RVI — Reverse interrupt, a positive acknowledgement, but is is also a request for the transmitting station to terminate the current transmission as there is a high priority message waiting to be sent.
ENQ — Inquiry, used to bid for the line in a point-to-point environment. In multipoint, it indicates end of polling or selection sequence. Also used to request retransmission of NAK or ACK if the response was garbled or not received when expected.
DLE — Data Link Escape. One use is in formatting WACK, ACK, or RVI as these are two-character codes. Example: ACK0 is DLE1 or 1031 in ASCII. Another use is to indicate transparent text mode. This mode is necessary to transmit binary files such as machine language programs, for some of the characters in this type of file can be bisync control codes. DLE STX start transparent mode, DLE ETX ends it.

There are a few other codes, but these will be enough for our example. For the first example, let's look at a point-to-point data link (figure 2). This type of data link is also

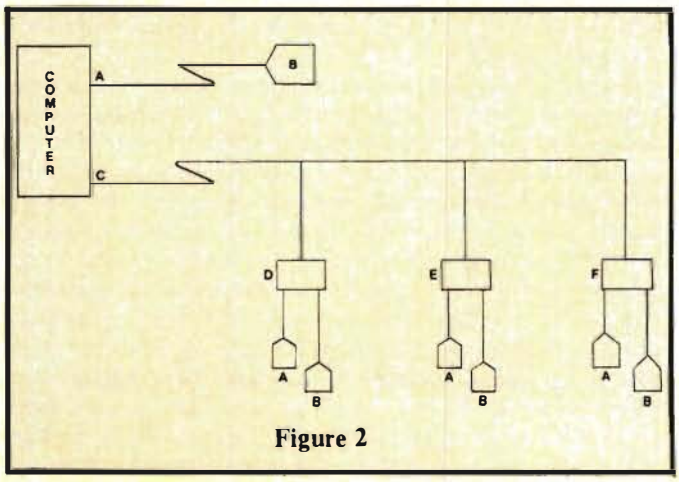


Figure 2

known as a "contention" system. I'll explain that in a moment. Before I do, let's get an idea of what the format of a message would be. I haved used the / to separate fields and the data flow will be from left to right.

/BCC/ETX/TEXT DATA/STX/HEADER DATA/SOH/

The header may or may not be present, but all other fields must be.

For our point-to-point example we will be transmitting between points A and B in figure 2. Let's start by sending a message from the computer "A" to the device at point "B." This could be some type of terminal or another computer. Location A will start by sending an ENQ asking "are you ready," B will respond with an ACK0. "A" will then send the message followed by ETX and appropriate BCC. If B received the message error free, it will respond with an ACK1, A can now send another message or send EOT saying "I have no more messages to send now."

If the message was received with an error being detected B would send a NAK and A would then have to re-send the message. This time, if it was error free, B would respond with an ACK1. If another error was detected the NAK sequence would continue a predetermined number of times, then A would send an EOT, and probably notify an operator via the system console that an unrecoverable error has occurred and would not try to send the message again until some command from the operator instructed it to do so. Once this command is entered the scenario would start over with an ENQ.

Now, suppose that after all messages were sent by location A and an EOT had been sent, A had another message to send. At this same instance, suppose B had a message to send to A and they both send an ENQ to each other. Interesting situation, if I do say so myself, for neither will respond with an ACK0 since ENQ is an invalid response to an ENQ. We now have defined why this type of link is sometimes called a contention system, for both have "bid" for the line and neither can get control. This situation only happens when both bid for the line at the same time. If point A had not bid for the line, B would have control and could send its messages in the same fashion described above for point A.

To get around this problem a concept termed "Master/Slave" is used. What this means is that the location designated as the control or master station will bid for the line in an interval that is less than the slave station, in this example point B. This enables point A to seize the line and send its messages prior to point B. Thus, we have eliminated this "contention" problem.

These are the basic concepts of a point-to-point data link using bisync regardless of the type of facility used, switched or private line. What about this new configuration called multipoint? The rules governing this type of configuration are different; let's see how.

Unlike point-to-point bisync, the remote terminals in a multipoint environment can never initiate the transfer of data, they can only send data upon an invitation to do so. Look at figure 2 again. I have included a multipoint circuit which contains four points. Point A again will be used to represent the computer, while points D, E and F could be anywhere from as nearby as the same building, or as far away as across country. Each remote point has an intelligent controller with two terminals each. We could have as many as 32 devices per controller, and as many as 32 controllers per circuit, however I doubt there would ever be a single circuit with that many devices (1,024) as I suspect response time for each transaction could, depending on the application, be unacceptable.

Each controller has all the intelligence to perform error recovery as described earlier, and has enough RAM for buffer space to handle all the devices attached to that controller. Now that we know something about each of these controllers, let's see how we will communicate with them.

In a point-to-point environment we saw that if one station wanted to send data to the other, it bid for the line by sending an ENQ. We can't do that with a multipoint circuit for we wouldn't know which device bid for the line, and when the computers bid for the line, each controller wouldn't know if we wanted to send data to it or one of the other controllers. How do we get around this? Well, the computer would solicit data from each location by invitation or "polling" as it is called. There are two types of polling we could do. One is called a "general poll," the other is a "specific poll." A general poll will ask a particular controller for data from any of its terminals, whereas a specific poll will ask a particular controller for data from a specific terminal. Let's see how that would work.

Suppose we wanted to see if controller "F" had any data from either of its two terminals. The computer would send the general polling characters for that location. Those characters for location "F" would be FF""ENQ, or 4646222205 in ASCII. Bisync always sends the station's address twice. I don't know why that is, just the way it was designed. That's why two Fs are sent. The two ""s indicate that it is a general poll and the ENQ simply asks "Do you have anything to send?" If it does, controller "F" would respond by sending the data with the appropriate framing characters such as STX, ETX and BCC. If the data was received error free, the computer would respond with an ACK-0; if an error was

detected it would respond with a NAK and the same error procedure would be followed as described for a point-to-point data link. If controller F had no data from either terminal, it would respond to the poll by sending an EOT.

Why didn't controllers "D" and "E" respond to the poll? Well, even though each of these controllers did see the poll on the line, they knew by the two Fs that the poll was not meant for them and simply ignored it. Had we wanted to poll either D or E, the Fs would have been replaced with the characters of the controller we wanted to communicate with.

Now that we have just received a message from controller F, we probably want to send some kind of reply for the operator. Maybe the message received was a request for the work location of an employee and we needed to get that information back to the terminal. To do this we would do what is called a "selection" sequence for that particular terminal on controller F. But how do we know which controller and which terminal on that controller to select? Well, part of the data we received from the poll were the codes that identify the controller and terminal that requested the information. The software in the computer would be aware of this and would save this information for the reply. Now that we know which terminal to select, how do we do it?

Assume we need to send the reply back to terminal A on controller F. The selection codes for controller F now must be different from those used for a poll so the controller will know what we are trying to do. The selection code for controller F is 57 in ASCII. That is the code for the letter W. The code for terminal A will be 41 in ASCII. Our selection sequence will then look like this: WW AAENQ, or 5757414105. If the controller was ready to receive the data for terminal A it would respond with an ACK-0 and the computer would now send the data. If it was received error free the controller would respond with an ACK-1. Again, if an error was detected by the controller, the same error procedure would be followed.

When do we use a specific poll? Well, this could be done instead of using a general poll, but it would take longer to service each terminal on a controller. Example: suppose controller D had 10 terminals. We would have to do a specific poll for the first nine before we got any data from the 10th one. That's not too efficient, particularly if the first nine had no data to send. Probably the most common use of a specific poll would be in the case where the host software determined that part of the data it received was incorrect, an invalid part number, etc., and sent a message to the terminal notifying the operator of this error. We would do a specific poll to receive this corrected information so we could process the transaction. Again, the specific polling characters would be unique to that device. For controller F, terminal A, the codes would be FFAAENQ or 4646414105 in ASCII. As you can see, this is similar to a general poll except we have replaced the two quotation marks with the terminal code.

One more example, then we will leave this topic. Suppose that device B on controller D has some kind of problem --perhaps this device is a printer and it has run out of paper. Also, assume we have just selected this printer for output. Obviously, we can't print without paper, so in response to the selection the controller will respond with an RVI. This RVI is a positive acknowledgement but it is also saying, "I have a problem with one of my devices, if you will poll me I will tell you what it is." The host software would then poll the controller and receive this status message. All status messages take priority over any other messages. In response to this status message the host would send an EOT or RVI. Now we can poll the terminal for input, but we still can't send any data to that printer until the error condition is cleared. How would we know when this happened? In response to some future poll, the controller would again send us a new status message saying "the problem has been cleared, you can now select that device."

As you can see, multipoint is a little more complicated than point-to-point; however, there are benefits that make this extra effort worth it. The major ones are probably line cost, and the saving of ports on the computer. In our example, we had six terminals on one circuit. If it wasn't for this multipoint capability, we would have to have six separate lines that would probably be more expensive than one multipoint line, plus we would need five additional ports to serve the other locations.

Now that we've seen how bisync protocol works once each point on the data link is ready to communicate, let's take a quick look at what happens prior to any data transfer taking place. To do this, I will be referring to Figure 1 again. The first two pins we want to look at now are pins 4 and 5, request to send and clear to send. Before the DTE will send any data it will first turn pin 4 ON to condition the modem for data transmission, and on a half duplex circuit to control the direction of data transmission. If the modem is ready it will turn pin 5, clear to send, ON. The DTE can now send its data. There is some delay in this RTS/CTS sequence. The actual amount of delay is dependent on the modem being used, but typically this delay will range from 50 to 250 milliseconds.

There are two more leads we should consider; those are pins 6 and 20, data set ready and data terminal ready.

Data set ready is used to indicate that the local terminal equipment is connected to a communications channel; it is not in any type of test mode; and, for switched lines, any timing functions required by the switching system to complete a call have been completed.

Data terminal ready is used to control switching of the data communications equipment to the communications channel. The ON condition prepares the equipment to be connected to the communication channel while the OFF condition causes the equipment to be removed from the communication channel following the completion of any "in process" transmission.

This should take care of the pins, or at least most of them, that are required to perform data communications. This interface is a standard and all types of data transmission may make use of the pins discussed, or at least part of them. As you know, we only need four of them for use with the Color Computer. For a complete definition of all the EIA leads there is a publication available from the Electronic Industries Association in Washington, D.C. The complete title of the document is *Interface Between Data Terminal Equipment and Data Communications Equipment Employing Serial Binary Data Interchange."*

So far, we have been discussing the use of modems, thus implying that our facilities, whether private line or dial-up are analog. We know that these analog facilities were originally designed to transmit voice and in order to transmit data we must convert the sending DTE's digital signal to analog and back to digital for the receiving DTE. This conversion process does take up some time no matter how minute this time period may be. Why not have facilities that are all digital and eliminate this conversion period? Well, such facilities are available. Let's take a brief look at some of them.

So far, we have been talking of speeds up to 9,600 BPS using modems and the various telephone companies' analog facilities. There is an offering by AT&T called Dataphone Digital Service, or DDS, that supports speeds of 2,400, 4,800 and 9,600 BPS without any additional equipment, except a device called a Data Service Unit that replaces your modems. This is an all-digital offering and is designed for synchronous transmission only on private lines. The interface at the DSU is the same standard RS-232-C interface as we found on modems. This service allows you to still use telephone company facilities and it is maintained by them.

> "... with fiber optics already being capable of transmitting at speeds over 400 million bits per second, who knows what tomorrow will bring?"

There are other digital services available; however, they all require special equipment to be installed at the user's locations. These offerings include microwave, satellite, AT&T's High Capacity Transport Service (HCTS), and Bell Labs' Fiber Optics. HCTS offers two speeds, 1.344 and

1.544 million bits per second. The other three offer speeds from 2,400 up to several million BPS. Also, DDS has a 56,000 BPS offering. Since we know there aren't many, if any, terminals that operate at speeds above 9,600 BPS, how can we make use of these speeds beyond that?

Let's take a look at Figure 3 and see how we can do this. What we have here is several ports from the computer going into something called a multiplexer. The multiplexer will take the data from each of these ports, combine the data into a single data stream and transmit all the data at a high rate of speed to another multiplexer which will separate the data and send it out on its ports onto the appropriate lines to the attached devices. The same is done for data coming from the remote network to the computer. In this example, I show a 56K BPS DDS data link between multiplexers. This data link could be any one of the other services as well. For

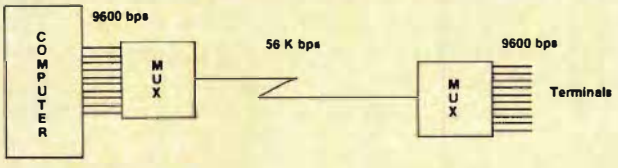


Figure 3

example, this link could be two earth stations transmitting to each other via a satellite.

By the way, if your computer network was all asynchronous you could make use of these high speed data links, for some multiplexers have the ability to convert the async data stream to synchronous and even perform error checking and recovery for you.

What is the advantage of using these types of services? First of all, we can see that these systems were designed for the large users. By using such a system they could be saving in line cost, for, instead of having several circuits they would only have the one high speed link. The time period to realize a savings will, of course, vary depending on the type of service providing the data link. Satellite earth stations and the service itself aren't cheap. Of course, there may be situations where you wouldn't have much choice as to which service to use. For example, if you had a data center in Los Angeles and one in Australia, you might have to use satellite services to get there.

As you can imagine, there are other configurations that we could use these high speed facilities for—let's not get into all of them here in any detail—but some more uses would be to transmit voice, facsimile, and of course, television broadcasting.

I hope it has been interesting to look at some ways to transmit data and to look at some of the new technology that is available today. And, with fiber optics already being capable of transmitting at speeds over 400 million bits per second, who knows what tomorrow will bring? I'm not saying you will ever have an earth station in your backyard, or a microwave tower on your roof, but just think; with the way technology is progressing and the cost of hardware continuously coming down, maybe one day when you log on to your favorite bulletin board with your home computer it may just be at 4,800 BPS or higher. Wouldn't that be nice?

C— LANGUAGE OF THE 80s

By Richard A. White

BASIC is a special type of computer language in that it is interpreted. This means that a machine language BASIC interpreter program must be resident in the computer to run a BASIC program. While a BASIC program runs, the interpreter takes each statement in turn, looks up what to do, and goes to the code in the BASIC interpreter that performs the required task. If you run a *FOR TO NEXT* loop that prints "X" 100 times, the interpreter must look up the *PRINT* token 100 times. This takes time. In a machine language program, either the address of the required code or the distance, in bytes, to the required code is in the program and the microprocessor goes directly to the address. However, getting or writing a program in machine language takes some doing. Most high level languages produce a machine code file. Some run directly. Others need some runtime code that interprets an intermediate code. Best is a stand alone machine code file that you can *LOAD, EXEC* and be in the program.

Writing in assembly language is one way to get to a machine language program. The assembly language source code is run through an assembler program which converts the assembly language to machine code also called the object code. Assemblers can be remarkably small, efficient programs. Radio Shack's EDTASM+ for the CoCo combines an editor, assembler and a monitor to run the machine program under operator control in less than 8K bytes.

Another approach is to write in a language whose source code is converted to machine language by a program called a compiler. Fortran, PL/1, and Pascal are all compiled languages though Pascal needs a runtime program to interpret the compiled file. *C* is unique in that it compiles to assembly language source code for the microprocessor in question and the assembly language file is then assembled to obtain the final machine code file. *C* is a small language so its compiler is small. A limited, but very useful *C* compiler fits easily into 16K. There now exist a number of *C* compilers for CoCo as well as a number of excellent references the beginning programmer needs to learn the language. Further, *C* has been implemented on many other micro and mini computers in such a way that *C* text is more transportable than most other languages. For these reasons, *C* makes a good choice for a next language to learn after BASIC.

C was written in 1972 by Dennis Ritchie at Bell Laboratories. Ritchie worked closely with Ken Thompson in developing the UNIX system. Nearly every software tool now supplied with UNIX including the operating system is written in *C*. Subsequently, *C* has just grown in use on its own merit. It has not had true believers out preaching the faith and bad mouthing competiting languages. It has not been "marketed." More like a carpenter choosing a good saw to quicker finish the job, programmers have chosen *C* so they can get good code to their client faster and get the paycheck. *C* feeds the family, if you will.

(Richard White has a long background with microcomputers and specializes in BASIC programming. With Don Dollberg, he is the author of the TIMS data base management program.)

C comes out of a family of languages whose traditions stress reliability, simplicity and ease of use. An international committee designed ALGOL 60 in 1960. Highly sophisticated, ALGOL has had major influence on programming language design but never caught on in the United States. Combined Programming Language appeared in 1963 as an attempt to "retain contact . . . with the realities of an actual computer." But, CPL was big and its hosts of features made it hard to implement and hard to learn. Basic Combined Programming Language was a 1967 attempt to boil CPL down to its basic good features. Ken Thompson at Bell Labs further simplified CPL to B to use on an early implementation of UNIX. But, BCPL and B were so condensed that they could only deal with certain kinds of problems. Building on these roots, Ritchie's *C* restores some lost generality without sacrificing simplicity or "computer contact." As for power, *C's* limited stock of parts can be assembled into more complex parts and combined into elaborate assemblies.

How does one get to the machine program using *C* on CoCo? Here are the mechanics without going into language details yet. First, an editor or word processing program is required that supports *all* the characters available from CoCo's keyboard when running in BASIC. Unfortunately, I thought that my beloved *Telewriter-64* did not make the grade since characters like the backslash did not seem to be available and are indispensible to *C*. I subsequently made the mistake of reading the instructions (nobody ever does that) and found the backslash is CLEAR-COMMA and was provided just so *Telewriter-64* could write *C* text. Two other needed characters were found by typing errors. These are "[" produced by SHIFT-@ and "]" produced by SHIFT-CLEAR. In the mean time, a surprisingly good editor, Mr Ed, was published in the June 1983 Rainbow, page 140. I modified the program to accept lowercase commands (*Bits and Bytes of Basic, Rainbow,* August 1983), modified the program name line to follow *C's* comment convention and have been very satisfied.

The *C* source code for the program is prepared, edited and saved to disk. Oh yes, *C* runs on disk only and gives a disk drive quite a workout. Cassettes just cannot do what has to be done. The *C* compiler is then loaded and executed. The compiler calls the *C* source code and converts it to assembly language source code which is considerably larger than the *C* text. During the process the *C* Compiler is reading in the *C* source from one disk file, converting it and saving assembly source to another disk file. Little of either *C* or assembler code is in memory at any time and you can combine a number of *C* source files to make a large assembly source file. Next, the assembler program is loaded from disk and executed. It goes to work on the assembly source file and you sit back while the disk chugs and chugs some more. I do believe the disk gets warm after a five to 10 minute session of constant rotation. The output object code is on disk now ready to *LOADM* and *EXEC*. If all went well you have a running machine language program. Since all *never, ever* goes well with *any* computer program in *any* language, the program will probably bomb and you will enter the debugging phase. Actually, the *C* compiler sends a variety of nasty messages which you will have responded to prior to assembling a corrected assembly source file so the problems now are ones of basic program logic and not ones of language systax. Now you can start to see one of the negatives of a compiled language compared to BASIC. When you write in BASIC, you need only *RUN* the program to see if the code you just entered works. With a compiled language, you have to compile your source code , and in the case of *C*, assemble it before you execute it.

> "*C* comes out of a family of languages whose traditions stress reliability, simplicity and ease of use."

C is a structured language. You need to declare your variables by type before you use them in each function that they are used. Further, variables are local to each function except those declared before "main()". This is somewhat like using DIM in BASIC in that it reserves space for that variable. Then you write your code in groupings called functions. These are analogous to subroutines in BASIC, but are named and are called by name. There are also standard functions, some of which are provided with the *C* compiler, which are not part of the language, but which you call from your program to perform certain tasks. *C* itself has no input or output statements. These are all in the standard library, provided with the compiler, and are written for the specific computer in question.

So, what does *C* source code look like? Following is a simple program that prints a table of Fahrenheit and Celsius temperatures. The table is not exact since the program performs integer math only.

```
/*   FAHR CELS.DAT      */
/*   print Fahrenheit-Celsius table for fahr=0, 20, ..., 300 */
main()
@
     char c;
     int lower, upper, step, fahr, celsius;
     lower = 0;       /*lower limit of table*/
     upper = 300;       /*upper limit*/
     step = 20;       /* step size */
     fahr = lower;

while (fahr <= upper) @
     celsius = (((5* 100)/9)*(fahr-32))/100;
     printf("%d      %d/n", fahr, celsius);
     fahr = fahr + step;
  $
  c=getchar();
$
#asm
  LIBS INCLIB.TXT
#endasm
```

First note that *C* is written in lowercase. The first two lines are comments enclosed by the /*. . .*/ delimiters called for by the compiler. Then main() defines the start of the program which is enclosed between the first @ and the last $. Next the variables are all declared, in this case as a character (char) and as integers (int). Variables can and should be meaningful words since this makes the source easier to read. Don't worry about space. None of the comments or variables get to the object code from the assembler. The "while (fahr<= upper)" is a control structure that tests if fahr is less than or equal to upper. As long as this is true, the following three lines between the @ and the $ are executed. The first of these lines calculates the value of celsius from the current fahr value. Then, both values are printed using printf().

Printf() is one of the standard library functions that is tailored to the particular computer. It is somewhat like *PRINT USING* in BASIC as it provides for some print formatting. The "%d%d/n" tells printf() to print a number, %d, print a few spaces, print a second number, %d again and then start a newline, /n. The variables containing the two numbers follow as fahr and celcius. In the library are a number of utility functions like printf(). Typically, the library file may be in a *C* compiler readable format and added to your program by the compiler or it may be in assembly language and added by the assembler. You can develop your own set of functions and keep them in a library to be added to your own programs or to be shared with other. Finally fahr is incremented by step (20) and looping continues until the test after "while" becomes false. Then this program goes to the next statement after the $ which is "c=getchar()". getchar() is another library function that works just like C=INKEY$ in BASIC. Here it serves to stop the program until a keyboard entry is made so one can read the table.

All function calls, even main() which is a function, are followed by parentheses where data to be sent the function is listed. In the printf() example, the peren's contents was

("%d%d/n", fahr, celsius). The printf() code got the full string of characters within the quotes and copies of the values in fahr and celsius. How printf() handles its work to get the values of fahr and celsius onto the screen is of no concern to the programmer. printf() is a black box that prints things to the screen in a way the programmer can control through what goes into the perens.

A *C* program is really a set of user written and library functions. Execution starts in main() where needed functions are called in their order of need. Ideally, main() will be as simple as possible with major blocks of logic in other functions which in turn call functions including themselves. This system makes *C* usable for a variety of programming tasks where needed statements are not in the kernel of the language. *C* has no math functions like log, sine, cosine, powers and the like. But, utility functions to do the tasks can be written and used just as the printf() function was used. Likewise *C* does not contain any string handling statements like RIGHT$, LEFT$ and MID$. Again, functions to do these things would need to be written or obtained from others who have already written them.

If, after all this, you are still interested in learning more, start with *The C Primer*, by Les Hancock and Morris Krieger, published by McGraw-Hill Book Company ($14.95). This and the other books mentioned below can be obtained generally off the shelf. Some computer stores will carry one or more of these. If the *Primer* turns you on, you will certainly need a copy of the bible. This is *The C Programming Language*, by Brian W. Kernighan and Dennis M. Ritchie, Prentice-Hall Software Series ($19.95). After the first chapter, Kernighan and Ritchie becomes so concise that the beginner finds the water quite muddy. But such conciseness is on the mark in a reference volume for the serious user. In addition this book is *the* definition of Standard *C* which is closely followed by most users and writers. Finally comes "The *C* Puzzle Book," by Alan R. Feuer, Prentice-Hall Software Series ($14.95). This book provides a series of short examples program for you to figure out with detailed analyses at the back.

> "A *C* program is really a set of user written and library functions."

But, it's no fun just reading and figuring while the computer sits cold on the table. Dugger's Growing Systems has an ad for their *C* compilers in this issue. I have the one for TRS-DOS which fits into a 16K machine. More memory may be needed for the text editor and text file, but not for either the compiler or the assembler. At this writing, Version 1.2 supports only the most necessary parts of the language. Version 2.0 implements another chunk and should be available by the time this is published. Duggers Compiler for FLEX is the 2.0 Version. Also coming from Dugger is a text editor written in *C* and an assembler written with assembly and linkage of *C* produced source code in mind. The Version 1.2 TRSDOS compiler was designed using Computerware's assembler. One of the attractions of this assembler is the ability to include LIBS assembler commands which will call additional assembly source files and combine these to produce a single object file. The other available *C* compiler for CoCo is from Frank Hogg Labs and also runs under FLEX. I have no information on this program. However, if it is an adaption of a compiler for another 6800 or 6809 based computer to the CoCo (one of the things FLEX makes possible) it may be the most comprehensive compiler for CoCo. Perhaps users will report their experiences.

By now you should know enough about *C* to decide to look further or not and know where to look should you choose. Obviously, we can only hope to touch a few bases in an article like this. To summarize and to help you calibrate your expectations should you go further, here are a few thoughts. If assembly language programming is low level and BASIC, PL/I, Pascal and others are high level, *C* is somewhere in the middle. A lot of the bells and whistles of a high level language are not there. While the BASIC programmer has access to the machine with *PEEK, POKE* and *VARPTR*, these are used with restraint. in *C* there is need to deal with addreses of data and manipulating pointers to data. As I read about *C* and its syntax logic, I think about how the computer is working more than I do when programming in BASIC. Finally, assembly language programmers should find *C* a natural step up the ladder since *C* permits assembly source inclusion within the *C* text. in the example above, #asm and #endasm enclose an assember command. They could just as well enclose assembly source code that the compiler passes directly to the assembler as part of the assembly source stream.

Communicating With Your Color Computer

By Bob Rosen

One of the added features of the Color Computer is the built-in RS-232 interface. It can be used to hook up a serial printer like the DMP-100. More important, though, it can be used to help you communicate. Communicate? That's right! There is a whole new world of telecommunications that many CoCo owners are unaware of. Besides playing games and writing programs, you can connect a modem to your RS-232 port. A modem is an electronic device which converts the computer's electrical pulses bit by bit into audible sounds that can be transmitted over the telephone lines. Modem stands for **MO**dulator/**DEM**odulator and there are two types available.

The two main types of modems currently on the market are the direct-connect modem and the acoustic-coupled modem. Both have their advantages. The direct-connect unit hooks right into the phone lines. For instance, Spectrum's Mini-Modem allows you to unplug your phone from the jack and hook it into our unit. Then you can plug the cable from the Mini-Modem back into your phone jack. This assumes that your phone and jack are modular. If not, Radio Shack sells two adapters, 279-393 and 279-351, that will convert your hook-up to modular. Since the direct-connect unit hooks right into the line, virtually no external noise can be detected which means someone else in the room can be watching TV or listening to a stereo. The acoustic-coupled modem is even easier to hook up. The attachment is the same as adding a telephone answering machine. If you have no extra jacks, Radio Shack sells two "Y" adapters, 279-357 and 279-370, for modular and four-prong systems. The acoustic-coupled modem is more susceptible to line noise as you are placing your phone's receiver into the unit's two cups, or "synthetic ears." This can be a problem if you are calling long distance and the signal on the other end is weak. Another problem is that today's modern phones might not fit the modem's circular cups (Slim line phones, for example). One advantage to this modem is that it is more flexible to use with multi-line phone systems than a direct-connect because all you have to do is push buttons to use the modem on another line.

(Bob Rosen is president of Spectrum Projects and has been running a successful CoCo mail-order business for a year and a half. He is ably assisted by Paul Rosen, Tom Delabo, Tom Genoski, Jerry Lato, Al Hartman, and Mel Silverman.)

The only other piece of hardware you will need for your modem is a serial cable to hook it up to the RS-232 port. For the Mini-Modem, you will need a 4-pin to BD25 cable. There are two other cables which can be very handy. One is a printer modem extension cable, which allows you to place your modem in another location. It extends your three-foot Radio Shack cable another fifteen feet. The second cable is the RS-232 expansion cable. This is a "Y" adapter cable which allows you to hook up your printer and modem inline, independently, at the same time. This eliminates constant swapping out of cables, since the Color Computer has only one RS-232 port. Also, as you will see later on, it will come in handy when using a smart terminal program.

Now that you have the hardware situation taken care of, you will need software. As with modems, there are two types of software. The first type is known as "dumb terminal software." Radio Shack sells a version under the Videotex name. It is catalog number 26-2222 and includes one free hour on the Dow Jones and CompuServe Networks. The software allows access to these networks, but that's about all it does. It lacks the desired features that a smart terminal program contains.

Now you might ask, what can a "smart terminal program" do for me that a dumb terminal doesn't? The smart terminal program can take the information you access from a telecommunications system and store it into your computer's memory. After you go off line from a system, you have the

option of printing it out or writing it as a cassette or disk file. Or, you can type a message off line and then upload with a single keystroke to the host computer.

Since networks such as CompuServe and The Source charge for their services, this can save you money as it can transmit it faster than you can type and it allows you to examine it for errors while off line.

Another nice feature of a smart terminal program is that it enables you to change any RS-232 or printer parameters. For example, it allows you to access half duplex systems and use 1200 Baud modems. You can run your printer at 1200 Baud and insert a carriage return after a linefeed.

Another bonus a smart terminal program has to offer is that you can scroll on line or off line up to 12 lines at a time. Where can you purchase such a program? Spectrum Projects sells a smart terminal package under the name *Colorcom/E.* (Other popular terminals include Super "Color" Terminal, by Softlaw; Color+Term+Plus, by Double Density; and Autoterm, by PXE Computing.) These do not offer any free time for CompuServe and Dow Jones, but do offer the desired features of a smart terminal program. One can still purchase from Radio Shack a Universal Sign Up Kit (26-2224) to go on line with CompuServe and Dow Jones. However, this kit contains no software. It can be used, though, in conjunction with a smart terminal program.

Now that you know more about the hardware and software which is involved, you may ask, "What else can I access besides large data bases which charge for the use of their systems?" The answer is a growing phenomena called Bulletin Board Services which are mushrooming all over the United States. A Bulletin Board Service, or BBS for short, consists of an individual with a personal computer and an auto-answer modem set up to have outside callers access their computer. (Yes, your own mini CompuServe!) Aside from the auto answer modem, one needs special BBS software to run the system. For example, Spectrum Projects owns and operates three BBS's which totally support the Color Computer. The three systems can be accessed by dialing (212) 441-3755, (212) 441-3766 and (212) 441-5719. These are operated and run on three Model IIIs with a Corvus 20 Meg hard disk drive. The system is free and no passwords are needed to access the system. Some features of the system are a message retrieval section, the Rainbow Tape section which contains over 300 programs from the pages of *the Rainbow*, a merchandise section, and a Rainbow Review section with over 100 evaluations of hardware and software. Using a smart terminal program, you can dial into the system and capture files from the download section. These files consist of games and utilities which you can run on your Color Computer. In the message retrieval section you can leave or retrieve messages with other users of the system. It is a way to get info on the Color Computer that is not readily available through other channels. You can ask technical questions, get opinions on Color Computer hardware or software, and place an ad. In the merchandise section is a list and description of products for your Color Computer.

There is one last thing you can do with your modem. Become a CoCo sysop! Sysop stands for **SYS**tem **OP**erator. The minimum requirements to run a BBS on a Color Computer are 64K of RAM, a disk drive, and an auto-answer modem. And of course, BBS software! (See *Rainboard* elsewhere in this issue for a complete free software program to start your own BBS.)

Bulletin Boards have proven to be very profitable investments as yours truly can attest to. Just imagine sitting in front of your CoCo BBS and watching it being accessed by a complete stranger at any time of day or night from anywhere in the world! Make way for the Bulletin Board System — the wave of the future!

About Your Subscription

Your copy of *the RAINBOW* is sent second class mail and, for subscribers in the United States, the date of mailing is printed on the label. If you do not receive your copy by the 25th of any month, send us a card and we will mail another immediately via first class mail.

You must notify us of a new address when you move. Notification should reach us no later than the 15th of the month prior to the month in which you change your address. Sorry, we cannot be responsible for sending another copy when you fail to notify us.

Your mailing label also shows an "account number" and the subscription expiration date. Please indicate this account number when renewing or corresponding with us. It will help us help you better and faster.

For Canadian and other non-U.S. subscribers, there may be a mailing address shown that is different from our editorial office address. Do not send any correspondence to that mailing address. Send it to our editorial offices at P.O. Box 209, Prospect, KY 40059. This applies to everyone *except* those whose subscriptions are through our distributor in Australia.

Stringing Along with String Variables and Functions

By Richard A. White
***Rainbow* Contributing Editor**

Color BASIC owners listen up. Most string handling is accomplished using Color BASIC Commands. So when we discuss strings and string functions, the only Extended BASIC one involved will be *LINE INPUT*.

Extended BASIC and Disk BASIC owners listen up. Did you know that the bulk of non-graphics and non-music programming is done with keywords and functions from the Color BASIC ROM? I wrote a program for *the Rainbow* that tests a program and reports back the number of Color BASIC, Extended BASIC and Disk BASIC Keywords and Functions used. I was suprised how few Extended and Disk commands were counted in programs that were designed to run from Disk or Extended BASIC. The message is that we can write many useful programs without Extended or Disk BASIC.

In fact, the Tutorial Program on Strings at the end of this column will run fine in Color BASIC. This is in response to those cards and letters to *the Rainbow* asking for more such programs. Tutorial programs help you learn in two ways. First, *RUN*ing the program helps some to focus on the elements of the lesson in a step by step manner a little easier that following text in a magazine. Secondly, those who type in the code will see multiple examples of what the tutorial is discussing. In either case, once you have the program on tape or disk, it will be easy to load it into CoCo sometime in the future and get a quick review of the subject.

We think of computing as dealing, for the most part, with numbers. There are some programming languages whose reason to be is to deal with numbers. While dealing with numbers is important, dealing with characters and words has become a major task for modern computers. Word processing and data base management, including mailing list data, client lists, employee information listings and inventory all work on the computerized handling of strings.

A string is a series of characters. Characters are anything not treated as a number. Letters, numerals, punctuation, control characters (such as carriage return and back space) and graphics codes are all just characters and are treated in the same way by BASIC. A string of characters does not have to make sense to still be a character string.

Characters can be entered into the computer in a number of ways. We will concentrate on program and keyboard entry, leaving data entry from cassette or disk to other columns. Strings may be part of a program. Here are two examples:

```
10 S$="DEMONSTRATION"
20 PRINT "THIS IS A "S$
```

In Line 10, the string "DEMONSTRATION" is assigned to the string variable S$. As we have said before, S$ does not equal the string. It represents the string and allows the computer to find the string.

Once BASIC runs Line 10 it knows where S$ string is and can use it anywhere in the program. In Line 20, there is also a string, "THIS IS A " that is not assigned to a variable. This is a string that BASIC uses when it comes to it in Line 20 and nowhere else. Note that BASIC also uses S$ in Line 20 to print THIS IS A DEMONSTRATION.

Strings can also be put in a program in *DATA* statements. In this case, the *DATA* must be read into string variables before it can be used.

```
10 FOR X=1 TO 4:READ S$(X):S$(X)+" ":NEXT
20 FOR X=1 TO 4:PRINT S$(X);):NEXT
30 END
1000 DATA THIS, IS, A, DEMONSTRATION
```

A *FOR...TO...NEXT* loop reads the the *DATA* into S$(X) string variables. We then add a space to each string. A *FOR...TO...NEXT* loop is used to print the strings and we get the same THIS IS A DEMONSTRATION printout we had before. Note that the program never reached Line 1000, but found the *DATA* and printed it all the same.

BASIC is set up to find the *DATA* wherever it is put in the program. The obvious place for *DATA* statements is at the very end of the program where BASIC can find the information, but does not waste time running over *DATA* lines.

I don't use *DATA* statements much. I like to define variables, including strings, at the end of program, send the program over these lines once and have them available from then on. Here is an example.

(Richard White has a long background with microcomputers and specializes in BASIC programming. With Don Dollberg, he is the author of the TIMS data base management program.)

5 GOSUB2000 : ...
**2000S$**="MESSAGE 1":S2$="MESSAGE 2":S3$= MESSAGE 3":RETURN

INPUT A$, *LINEINPUT* A$ and I$=*INKEY$* all seek a keyboard input that is put into a string. *INPUT* A$ accepts characters from the keyboard for entry into a string until it sees a carriage return [ENTER], quote, comma, or semicolon.

This is good news and bad news. The good news is that you can enter a number of strings with one *INPUT* statement. The bad news is the obvious limits on what the string you [ENTER] can contain. *INPUT* also lets you print a string to the screen, telling the operator what to enter.

100 INPUT "ENTER YOUR NAME AND TELEPHONE NUMBER";NA$,TL$

"The big value of *LINEINPUT* is in word processing and data base management programming where all keyboard characters need to be entered into strings."

The Extended BASIC statement *LINEINPUT* A$ allows entry of any keyboard character except the carriage return [ENTER], which terminates the string. It allows only one string to be entered at a time and does not support printing a string after the keywords. The equivalent of the above INPUT code follows.

100 PRINT"ENTER YOUR NAME AND TELEPHONE NUMBER" : LINEINPUT NA$: LINEINPUT TL$

The big value of *LINEINPUT* is in word processing and data base management programming where all keyboard characters need to be entered into strings.

In the ideal program, all operator choices will be entered as single keystrokes. Why enter a character and an [ENTER] when we can do the same thing with one keystroke. INKEY$ is our hero.

20 I$=INKEY$:IF I$="" THEN 20 ELSE RETURN

This subroutine is a real workhouse. Put it early in the program so BASIC can get to it in a hurry. You can then call it from various places in the program. A good example in the tutorial is Line 18, which lets the user pace himself through the program.

Once the character or string is entered, it is processed or used in some way. BASIC includes a variety of powerful processing facilities.

Strings can be combined or added to each other in a process called concatenation.

```
5 GOSUB 2000
100 A$= B$ + C$ + D$ : PRINT A$ : END
2000 B$="THIS ": C$="DEMOS ": D$="CONCATE-
NATION.": RETURN
```

Here three strings were added to form a new one, A$, which was then printed. This becomes a powerful tool when you need to make a string that includes characters not available from the keyboard. There is no other way to make a string in a program that contains a quote than to add it to a string using CHR$(34). In the tutorial program, I need to print to the screen example lines of BASIC that define strings. How would you write BASIC that prints the following:

```
10 A$="TEST STRING"
```

The First try might be:

```
10 PRINT"10 A$ = "TEST STRING""
```

But that won't work since BASIC will read the second quotation mark as the end of a string. It will think that TEST is a variable, which has not been evaluated, of course, and print a zero. It treats STRING the same way. The final "" is thought of as a null string, a string with nothing in it which prints nothing. If you run line 10 you get:

```
10 A$=0 0
```

The ASCII value of " is 34. Let's rewrite Line 10 to tell the program to print CHR$(34), the character whose ASCII value is 34 whenever we need to print a ". We will put the whole thing into a new string so we can easily print it any time we need.

```
5 A1$ ="10 A$=" + CHR$(34) + "TEST STRING" +
CHR$(34)
10 PRINT A$
```

This will do the job. Key it into your CoCo and prove it to yourself.

What CoCo can put together, CoCo can take apart. Taking strings apart, or getting pieces of strings, is fully as important as putting strings together. Color BASIC offers three functions that return a part of a string.

```
L$ = LEFT$ (A$,NUMBER)
M$ = MID$ (A$,POSITION, NUMBER)
R$=RIGHT$(A$,NUMBER)
```

In all cases, A$ is the string we need to get something from or take apart. NUMBER is simply the number of characters from the left end of A$ that need to be put into L$. A$ remains unchanged, and the characters in question are in both A$ and in L$. When *RIGHT$* is used, it works the same way except it counts back from the right end of A$ to determine which characters to get.

*MID$* is more general, and more powerful, but has only one more parameter to remember. POSITION is the number of characters from the left end of the string where BASIC is to start getting characters. NUMBER is the length of the string to get and put into M$. Did you happen to think that we can throw *LEFT$* and *RIGHT$* away and use *MID$* for all string disassembly? The substitution for *LEFT$* is simple to use 1 for position.

```
L$=MID$ (A$, 1, NUMBER)
```

This will work exactly like *LEFT$* above.

To substitute *MID$* for *RIGHT$*, we need to know how long A$ is and subtract NUMBER from that length to get the starting position.

```
100 L=LEN(A$): R$= MID$(A$, L-NUMBER, NUMBER)
```

Looks simple, doesn't it? Well it is and it isn't. The problem is not in knowing how to use *LEFT$*, *RIGHT$* and *MID$*. It comes when you need to write the program to calculate POSITION and NUMBER. This I cannot simply lay out in a tutorial. It depends on the logic of the program you are writing. Indeed, string handling functions are basically simple. It is how they are used that gets complicated. Do not despair! Many before you have learned and come to appreciate the power of BASIC strings.

Custom Interface For The *Gemini-10*

By William R. Hall

Interfacing the CoCo to a *Gemini-10* line printer has not been an easy path to tread. From bulky serial-parallel converters to factory interfaces, there's been a price to pay, if not in dollars, then in transmission speed. The interface presented in this article can be built for less than $20 (complete kit); and transmission speeds are selectable from 110 to 9600 Baud. We call it the 8010 interface.

Before you decide to upgrade or backup a current interface, let's cover a few bases. This interface is only for the *Gemini-10* or *15*, and not the *10X*. The 8010 uses the intelligence of the printer, but does not require any modification of the printer or CoCo. If you use the printed circuit in Figure 1, you can mount the 8010 in the same place as the factory interface and still have access to the set-up switches without removing the case. All options and buffering of the Gemini remain functional. In short, the 8010 emulates the Star Micronics serial interface for the signals required by the CoCo.

Circuit Theory

Understanding the circuit theory is not necessary to build the interface, but for those who wish to know . . . read on. The Gemini is capable of handling serial inputs, providing a status word is read to set up the microprocessor for this function. The 8-bit status word is strobed into the microprocessor on pin 8 of CN2. All timing of this word is also managed by the processor through pin 5 and pin 6 of CN2. Table 1 shows the function of the individual bits in the status word. The SN74165 is used to create this status and permit it to be retrieved by the Gemini's electronics.

The remaining electronics are buffers and signal conditioning circuits. Data from pin 4 of the serial port of the CoCo is inverted and converted to TTL levels and then fed to pin 3 of CN2. The system busy signal (pin 1 of CN2) is buffered and then sent to CoCo to stop transmission when the Gemini's buffers are full or off line. "System Busy\" is also fed back through pin 9 of CN2.

Power is supplied by the printer through CN3 in the case of a *Gemini-10,* or CN12 with a *Gemini-15*, (see Figure 2). Only +5V and ground are required to interface the CoCo.

Using The Printed Circuit

The circuit board shown in Figure 1 has been designed to mount inside the printer, directly behind the serial port panel. All components and cables are mounted on side 1, except the switch bank and data cable. This is done to enable the switches to be accessed through the serial port panel. The vacant IC position and the diode pads are reserved for a 68188 line driver, for installation over 50 feet between host and printer. Jumping pins 8 and 9 of this vacant slot will enable operation without the line driver. Pin numbers on the upper set of pads correspond to CN2, and the lower set correspond to the serial port of the CoCo. The personality jumpers shown in Figure 1 are for operation with the CoCo.

Step By Step

1) install IC 1 and 2
2) install R1
3) install SW1 on side 2
4) connect ribbon cable pads 1—9
5) connect wire to pads +5 and G
6) solder jumpers as in Figure 1
7) connect data cable (3 wire) to the pads next to SW1 on side 2 and to 4-pin din
8) install connectors on cables
9) use small mounting brackets and 6mm x 3mm screws for mounting
10) set switches (see Table 2)

Parts List

IC1 . . . . . SN74165
IC2 . . . . . SN7404
R1 . . . . . 1K ohm @ ¼ W
SW1 . . . . . 275-1301 Radio Shack
CN2 . . . . . CE10028-10 (Panduit)
Serial Port Plug . . . . . 274-008 (Radio Shack
P.C.B. . . . . SN6-83 (Dayton Assoc.)

Table 1

Bit 1	irrelevant
Bit 2	parity (on,off)
Bit 3	serial select
Bit 4	serial select
Bit 5	parity (odd,even)
Bit 6	Baud rate
Bit 7	Baud rate
Bit 8	Baud rate

Table 2

Baud rate	SW2	SW3	SW4	Poke loc.
				150
9600	ON	ON	ON	1
4800	ON	ON	OFF	7 *
2400	ON	OFF	ON	18
1200	ON	OFF	OFF	41
600	OFF	ON	ON	87
300	OFF	ON	OFF	180

* Recommended

Parity	SW6	SW1
no parity	OFF	OFF

Serial Select	SW7	SW8	SW5
	ON	OFF	OFF

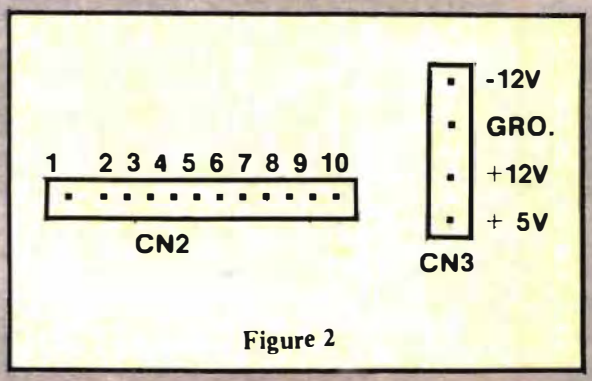


Figure 2

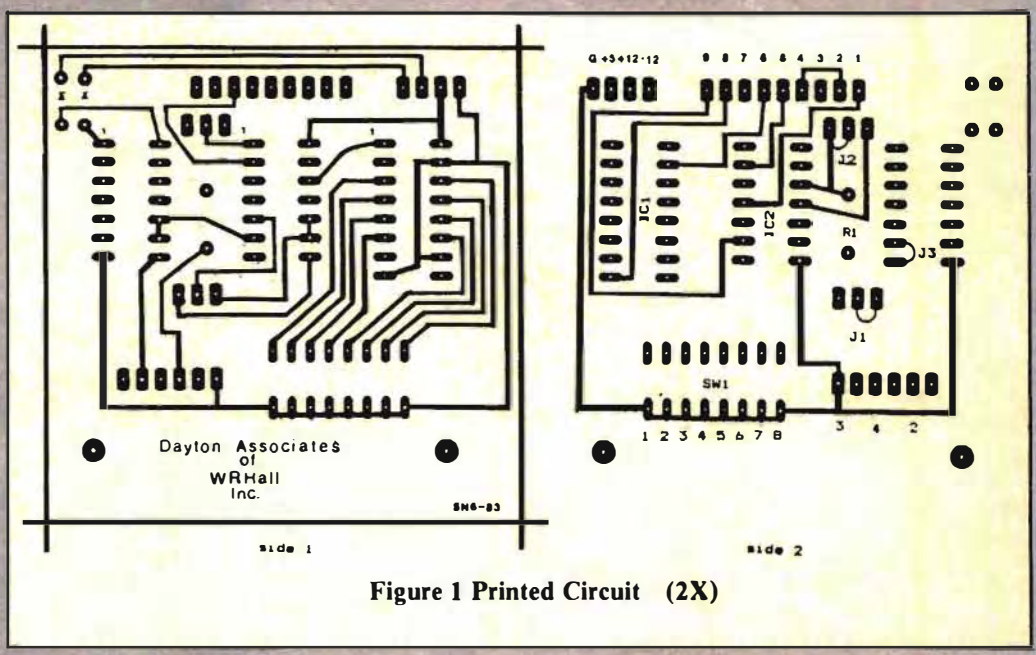
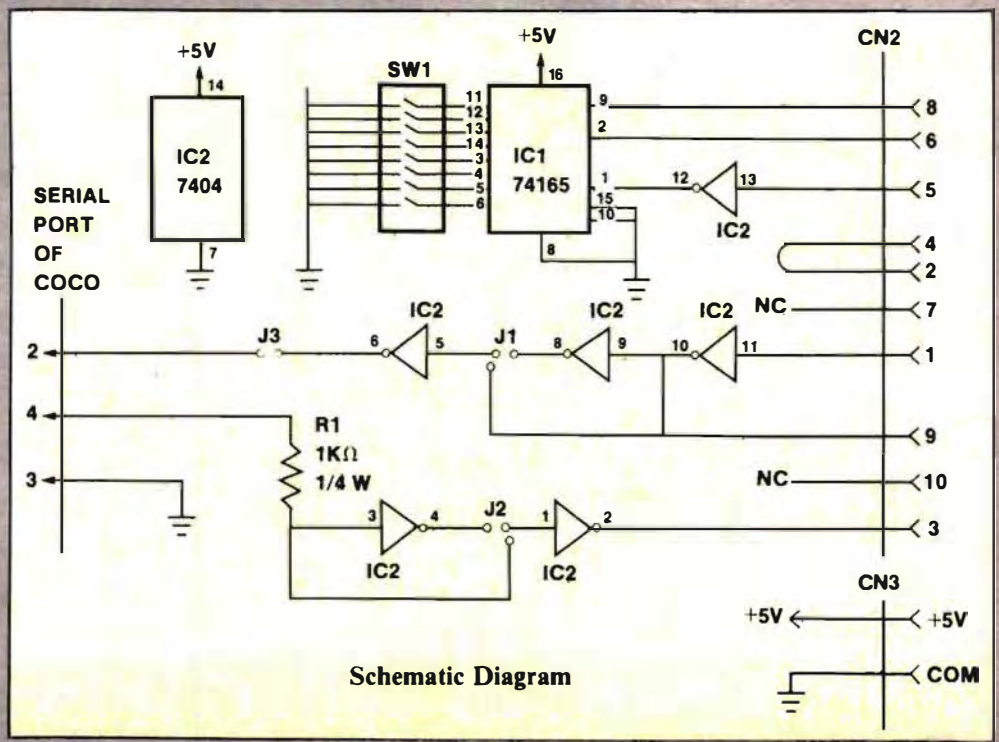


Figure 1 Printed Circuit (2X)



Schematic Diagram

How does your computer read your mind?

How Is How!

Here is a cute number guessing game sent to us by Henry Portela of Martin, Tenn. Called *How*, it requires 16K and Extended Color BASIC. Although it requires hardly any effort on the user's part, it is fun, especially for children. The most impressive aspect of *How* is the hi-res graphics used in the title page and throughout the game. The title page is done in perspective, giving it a three-dimensional look.

After typing and *RUNning* the program, press R to start the game or *ENTER* to see the instructions. You will then be asked to pick a number between one and 100 in your mind. You will be presented with an array of numbers and asked if your secret number is among the cluster. Respond by pressing Y (yes) or N (no). The CoCo will repeat this procedure several times, using different number clusters. Then your CoCo will "read your mind" and tell you your secret number.

Program Line Description

Line	Comments
1	speed up the action with *POKE* 65495,0
2	clears string space
3—6	letters and numbers data
7	*GOSUB*46—draws title page, circles and sound, waits for R to start the game, or *ENTER* for information *GOSUB*68—draws information and waits for R to start game
11	transforms the words in drawings for the screen
12—30	main loop
31	transforms all numeric data in drawing for the screen
39—46	numeric data
47—111	part of main routine or loop

Henry included a high speed poke, so remember to *POKE* 65494,0 before *CSAVEing*.

How does not require skill or a certain amount of intelligence. It's just a very simple math trick with brilliant graphics.

(Henry Portela, a foreign student residing in Martin, Tenn., came to the United States four years ago "without knowing a word of English." He has had his CoCo for only two years and this is his first submission to ***the Rainbow.****)*

```
7 ..... 0352
38 ....068C
46 ... 0A7B
67 .... 0E74
92 .... 1205
END .. 1410
```

The listing:

```
1 POKE 65495,0
2 CLEAR 200:DIMO$(26),H$(10),N(5
2),Y(7):RESTORE
3 PMODE 1,1:PCLS:CLS0:PLAY"V30L2
55T255":H$(0)="U10R10D10L10R10BR
4":H$(1)="R10L5U10G5BF5BR9":H$(2
)="U5R10U5L10BD10R10BR4":H$(3)="
R10U10L10R10D5L8R8D5BR4":H$(4)="
BU5U5D5R10U5D10BR4":H$(5)="R10U5
L10U5R10BD10BR4"
4 DATAU4E4F4L8R8D4BR4,U8R7FD2GL7
R7FD2GL7R7BR5,U8R8L8D8R8BR4,U8R7
F1D6G1L7R7BR5,U8R8L8D4R6L6D4R8BR
4,U8R8L8D4R6L6D4BR12,U8R8D2U2L8D
8R8U4L4R4D4BR4,U8D4R8U4D8BR4,R4U
8L4R8L4D8R4BR4,U2D2R8U8D8BR4,U8D
4R4E4G4F4BR4,U8D8R8BR4,U8F4E4D8B
R4,U8F8U8D8BR4
5 DATAU8R8D8L8R8BR4,U8R8D4L8D4BR
12,U8R8D8L8R6H2F4H2R2BR4,U8R8D4L
8R4F4BR4,R8U4L8U4R8BD8BR4,BR4U8L
4R8BD8BR4,U8D8R8U8D8BR4,BR4H4U4D
4F4E4U4BD8BR4,U8D8E4F4U8D8BR4,E8
G4H4F8BR4,BR4U4H4F4E4BD8BR4,E8L8
BD8R8BR4
6 FORT=1 TO 26:READ O$(T):NEXT T
:PMODE 1,1:PCLS:H$(6)="U10R10L10
D5R10D5L10BR14":H$(7)="E10L10BR1
4BD10":H$(8)="U10R10D10L10U5R10D
```

```
5BR4":H$(9)="R10U10L10D5R10D5BR4
"
7  GOSUB 46
8 GOSUB 68
9 PMODE1,1:SCREEN1,1
10 GOTO 12
11 L=LEN(R$):FOR TY=1 TO L:M=ASC
(MID$(R$,TY,1))-64:IF M=-32 THEN
 DRAW"BR10":NEXT TY:RETURN ELSE
DRAWO$(M):SOUND 50,1:NEXT TY:RET
URN
12 DRAW"S4BM2,2BD8"
13 Y(1)=64:Y(2)=2:Y(3)=16:Y(4)=1
:Y(5)=32:Y(6)=8:Y(7)=4
14 FOR S=1 TO 7 :T=Y(S)
15 FOR X=1 TO 52:READ N
16 A$=STR$(N)
17 C=C+1:GOSUB 31
18 IF C=6 THEN DRAW"BL254":DRAW"
BD20":C=0
19 IF A$=STR$(0) THEN 21
20 NEXT X
21 DRAW"S4C4BM0,190":R$=" IS YOU
R NUMBER HERE ":GOSUB 11
22 E$=INKEY$:IF E$=""THEN 22
23 IF E$="Y"THEN TT=TT+T:GOSUB 3
3 :GOTO 28
24 IF E$="N" THEN TT=TT:DRAW"BM2
,2BD8":GOSUB 33 :GOTO 28
25 IF E$="Q"THEN PCLS:RUN
26 IF S=7 THEN GOSUB 33:GOTO 27
27 GOTO 22
28 C=0:PCLS:DRAW"C4BM2,2BD8"
29 SOUND S+150,S+1:NEXT S
30 GOTO 30
31 DRAW"C3BR2BU5R4BD5BR4":FOR L=
2 TO LEN(A$):M=VAL(MID$(A$,L,1))
:DRAW H$(M):PLAY "L10003C02AE":N
EXT L
32 RETURN
33 IF S=7 THEN  34 ELSE GOTO 36
34 IF TT>101 THEN TT=0
35 IF S=7 THEN A$=STR$(TT) :PCLS
:GOSUB 94:GOSUB 103:DRAW"S8BM80,
120":GOSUB 31 :GOSUB 98:GOSUB107
36 RETURN
37 GOSUB 94
38 GOTO 38
39 DATA 64,65,66,67,68,69,70,71,
72,73,74,75,76,77,78,79,80,81,82
,83,84,85,86,87,88,89,90,91,92,9
3,94,95,96,97,98,99,100,0
40 DATA 2,3,6,7,10,11,14,15,18,1
9,22,23,26,27,30,31,34,35,38,39,
42,43,46,47,50,51,54,55,58,59,62
,63,66,67,70,71,74,75,78,79,82,8
3,86,87,90,91,94,95,98,99,0
41 DATA 16,17,18,19,20,21,22,23,
```

```
24,25,26,27,28,29,30,31,48,49,50
,51,52,53,54,55,56,57,58,59,60,6
1,62,63,80,81,82,83,84,85,86,87,
88,89,90,91,92,93,94,95,0
42 DATA 1,3,5,7,9,11,13,15,17,19
,21,23,25,27,29,31,33,35,37,39,4
1,43,45,47,49,51,53,55,57,59,61,
63,65,67,69,71,73,75,77,79,81,83
,85,87,89,91,93,95,97,99,0
43 DATA 32,33,34,35,36,37,38,39,
40,41,42,43,44,45,46,47,48,49,50
,51,52,53,54,55,56,57,58,59,60,6
1,62,63,96,97,98,99,100,0
44 DATA 8,9,10,11,12,13,14,15,24
,25,26,27,28,29,30,31,40,41,42,4
3,44,45,46,47,56,57,58,59,60,61,
62,63,72,73,74,75,76,77,78,79,88
,89,90,91,92,93,94,95,0
45 DATA 4,5,6,7,12,13,14,15,20,2
1,22,23,28,29,30,31,36,37,38,39,
44,45,46,47,52,53,54,55,60,61,62
,63,68,69,70,71,76,77,78,79,84,8
5,86,87,92,93,94,95,100,0
46 REM
47 PMODE 3,1:PCLS
48 DRAW"BM8,123;E40R16ND12G16R12
NE4R12E16R16ND12G40D12NE12L16U12
NR16E16L24G16L16D12R16NU12E16NU1
2R12"
49 DRAW"BM76,123;E40R12F40D12L92
U12NR92R16BE8E16ND12R12ND12F16L4
4R12E4R12F4"
50 DRAW"BM180,123;H40ND12R16F32R
12H16ND12R12H4U12R16F40D12L68NH1
2U12NR68R16BH8R12BR12NH16R12H32"
51 PAINT(16,128),2,4:PAINT(60,12
8),2,4:PAINT(104,128),2,4:PAINT(
200,128),2,4:PAINT(32,120),3,4
52 PAINT(56,96),3,4:PAINT(48,116
),3,4:PAINT(80,116),3,4:PAINT(10
8,108),3,4:PAINT(120,108),3,4:PA
INT(136,110),3,4:PAINT(160,108),
3,4
53 PAINT(190,112),3,4:PAINT(200,
100),3,4
54 PAINT(40,100),4,4:PAINT(108,1
00),4,4:PAINT(160,100),4,4
55 PMODE 4,1:SCREEN1,1
56 FOR I=1 TO 20STEP 10:CIRCLE(1
24,20),I,3,.1:NEXT I
57 FOR I=1 TO 40 STEP 8:CIRCLE(1
24,23),I,3,.1:SOUND 10+I,1:NEXTI
58 FOR I=1 TO 60 STEP 10:CIRCLE(
124,30),I,3,.1:SOUND150+I,1:NEXT
 I
59 FOR I=1 TO 80 STEP10:CIRCLE(1
24,40),I,3,.1:SOUND180+I,1:NEXTI
60 FOR I=1 TO 100 STEP10:CIRCLE(
```

```
124,52),I,3,.1:SOUND 180-I,1:NEX
TI
61 FOR I=1 TO 120STEP10:CIRCLE(1
24,70),I,3,.1:SOUND 140-I,1:NEXT
I
62 FOR I= 1 TO 160 STEP10:CIRCLE
(124,152),I,3,.1:SOUND200,1:NEXT
I
63 FOR I=1 TO 180 STEP10:CIRCLE(
124,180),I,3,.1:SOUND150,1:NEXTI
64 E$=INKEY$:IF E$=""THEN 64
65 IF E$=CHR$(13) THEN PCLS:RETU
RN
66 IF E$="R"THEN PCLS:GOTO 9
67 GOTO 64
68 REM
69 PMODE 1,1:PCLS2:SCREEN1,0
70 DRAW"S12C4BM50,40":R$="H O W"
:GOSUB 11
71 DRAW"S8C3BM2,80":R$="IS A SIM
PLE":GOSUB 11
72 DRAW"S8C3BM30,110":R$="T R I
C K":GOSUB 11
73 DRAW"S8C3BM100,140":R$="OF":G
OSUB 11
74 DRAW"S8C4BM76,170":R$="MATH":
GOSUB 11
75 FOR X=1 TO 2000:NEXT X
```



```
76 PCLS:SCREEN1,1
77 DRAW"S6C3BM0,40":R$="HOW TO P
LAY HOW":GOSUB 11
78 DRAW"S4C4BM50,60":R$="YOU WIL
L THINK ":GOSUB11
79 DRAW"S4C4BM40,80":R$="A NUMBE
R BETWEEN":GOSUB 11
80 DRAW"S4C4BM10,100":R$=" ONE A
ND ONE HUNDRED":GOSUB 11
81 DRAW"S4C4BM2,120":R$="THE CO
CO WILL ASK YOU":GOSUB 11
82 DRAW"S4C4BM10,140":R$="  IS T
HE NUMBER HERE":GOSUB 11
83 DRAW"S4C4BM10,160":R$=" YOU L
L ANSWER Y OR N":GOSUB 11
84 FOR X=1 TO 3000:NEXT X
85 PCLS
86 DRAW"S8C3BM60,20":R$="T H E N
":GOSUB 11
87 DRAW"S4C4BM0,60":R$="THE MAGI
C OF THE CO CO":GOSUB 11
88 DRAW"S4C4BM40,80":R$="  WILL
GIVE YOU":GOSUB 11
89 DRAW"S4C4BM30,100":R$=" THE R
IGHT ANSWER":GOSUB 11
90 DRAW"S4C2BM10,160":R$="IF YOU
 WANT ":GOSUB 11
91 DRAW"S4C2BM10,180":R$="TO PLA
Y AGAIN PRESS R":GOSUB 11
92 GOTO 64
93  PCLS:RETURN
94 DRAW"S4C3BM30,40":R$="DO YOU
BELIEVE IT":GOSUB 11
95 DRAW"S4C3BM20,60":R$="   THE
CO CO SAYS ":GOSUB 11
96 DRAW"S4C3BM20,80":R$="   YOUR
 NUMBER IS":GOSUB 11
97 RETURN
98 DRAW"S4C4BM20,140":R$=" HA HA
 HA THE CO CO ":GOSUB 11
99 DRAW"S4C4BM10,160":R$="  CAN
READ YOUR MIND":GOSUB 11
100 DRAW"S4C3BM10,175":R$="  DO
YOU WANT TO PLAY":GOSUB11
101 DRAW"S4C3BM90,190":R$="PRESS
 R ":GOSUB 11
102 RETURN
103 FOR X=80 TO 90
104 SOUND X,1:SOUND X+10,2:SOUND
 X-10,1
105 NEXT X
106 RETURN
107 E$=INKEY$:IF E$=""THEN 107
108 IF E$="R"THEN PCLS:RUN
109 IF E$="N"THEN POKE 65494,0:E
ND
110 IF E$=CHR$(13)THEN PCLS: GOT
O 1
111 GOTO 107
```

Copyrighting Your Software
Part 3

By Tom Nelson
Rainbow Contributing Editor

That crispy, fresh certificate from the Copyright Office sure looks good on your wall. So what does your copyright give you. What are your rights and what are your obligations?

Yes, obligations. Your copyright is a bundle of rights created by Congress which has certain requirements to be retained. You must follow the technical requirements of the law to obtain and retain your copyright.

Let's first investigate the rights you get with your copyright, registered or not. Section 106 of the Copyright Act gives the owner of the copyright exclusive rights to do and to authorize several things:

1) Reproduce the copyrighted work in copies;
2) Prepare derivative works based upon the copyrighted work;
3) Distribute copies of the copyrighted work to the public by sale or other transfer of ownership, or by rental, lease or lending.

Well, there you go. You get the right to control the sale and manufacture of your computer program, the translation of your program for use on other computers and the copying of your program. You get to reap the benefits of your creativity — for awhile.

Your copyright is not unlimited in duration. Still, for the purposes of the viability of your program in the marketplace you have a virtual monopoly. Your copyright is good for your lifetime plus fifty years, unless you are the owner of an anonymous or a pseudonymous work, or a work made for hire. Then the copyright is good only for seventy-five years from the date of first publication. That's a long time with computer programs. Even the Color Computer will be long gone by that time, and your computer program will be antique and quaint at best (yes, even the *VIP Writer*™).

These rights are great and give you virtual control over your program. I say virtual because the copyright law includes certain exceptions to your exclusive rights. One limitation is that there are certain "fair uses" of the program. Section 107 of the Copyright Act states that uses of copyrighted works for purposes such as criticism, comment, news reporting, teaching, scholarship and research is not infringement of copyright. As you can see, the "fair use" exception is rather limited, and applies only to a very limited extent to computer programs, since only rarely would a critic or scholar ever look at your object code.

This and other minor exceptions to your copyright rights are insignificant compared to the giant exception created by the 1981 amendment to the Copyright Act for computer programs. The 1981 amendment added a special section 117 to apply to computer programs:

> Notwithstanding the provisions of section 106, it is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:
>
> 1) that such new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner; or
> 2) that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.
>
> Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer or all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

This section is potentially devastating to the author of a program and perhaps even to the microcomputer software industry as a whole. It allows the owner of a copy of your program to legally make an archival copy and legally alter

"Your copyright is not unlimited in duration. Still, for the purposes of the viability of your program in the marketplace you have a virtual monopoly."

(Tom Nelson was formerly a special assistant attorney general for the State of Minnesota. He currently is general counsel for Softlaw Corporation, makers of the ***VIP Library***™*, and of* ***ColorQuest***™ *games.)*

your program to work with the owner's special use.

This section was added to the Copyright Law after considerable study, and after a special committee investigated the relation of copyright law to computers. Unfortunately, this committee was comprised of specialists who were only familiar with the computers of the time, that is, mainframes. In the short time from the late 1970's to the present, micros have become of unprecedented importance. But the problems of micros are not identical to those for mainframes, and the solutions differ considerably.

Mainframe computers are generally of limited quantity and are programmed with high-level language programs. Software for mainframes is sold in very limited quantities and for very high prices, such as $100,000. The programs are usually customized for the particular use and company, either by the program designer or the company itself. Thus, to allow copies and customization by the program owner applies quite logically to programs written for the limited number of mainframes. The investment of such a large sum of money, and the special application for which the program must be designed, make it imperative that an archival copy of the program is obtained and the right to modify subsist.

Micros, a phenomenon new to the computer industry, have lead to a different software situation. Software for micros is generally not meant to be altered by the end user. Alteration assumes a specialized use, a concept alien, by and large, to the micro market. Micros are mostly used for generalized purposes, for example, word processing, data management and communications. Moreover, software for micros is very cheap, and easily replaceable. Thus, the logic which allows a company to make a copy to protect its $100,000 investment does not apply to the inexpensive micro market.

Still, the law exists, and has caused a considerable disruption of the software market. It still allows the end user to make copies and make alterations. If limited to the skills of users this would not be a problem.

The problem arises from the provision allowing people to legitimately make archival copies. This "loophole" allows companies to sell "copy" programs to allow end users to make "backups" of their programs. If the end user had no such right, copy programs would clearly be illegal as instruments only for infringement.

It is obvious to all that these so-called "backup" programs are not just used by the honest to make backups of their programs, a practice approved of by all; they are used by all sorts of persons to steal programs. Of course, the makers of such copy programs include all sorts of disclaimers, but the user often is not mindful of these when in the privacy of his or her own home. The temptation to avoid even the smallest amount of money by just making a copy is just too great.

The ability of any Tom, Dick or Jane to steal programs rather than purchase them will eventually have two results. It will inhibit software houses from producing programs for a particular computer, and it will substantially decrease the quality of programs available. There is little incentive to put months, even years, of research into a program when you know that only 30 percent of all copies of your programs used by people will have been legally purchased.

So the law, as it stands, will eventually lower the quality of software. Obviously, for the benefit of all concerned, the law has to be changed for the micro market to prohibit the end user from making archivals, while perhaps requiring archival copies to be supplied by the manufacturer. You, as an author, should do what you can to make sure that the

section 117 is suitably changed to protect your interests in your creations.

What a digression! You've now got an idea of your rights. Now, how to use them. As owner of the copyright you have a few options to make the most of your program. You can market your program yourself, you can license your program for marketing by another company, or you can totally assign your rights in the program.

Marketing your program yourself gives you two options. You can either sell copies of your program, or you can license the use of your program. Licensing theoretically gives you more control over the use of your program. I'm sure you've seen these types of license agreements. For example, Tandy uses them. With a license, the end user is not buying a copy of the program, but paying for the right to use the programs under the conditions set out in the license agreement. Thus, the licensee could be prohibited from making archivals or altering the product in any way.

It sounds wonderful, almost too good to be true. Well, it is. Licenses work well when there is a negotiated agreement or an atmosphere of contractual agreement. These do not, however, typify the merchandise market where software is "sold." Instead, software is "sold" like apples or records. You go in, see the package and buy it without any thought that there may be some strings attached. In fact, you think that you are buying the software, not licensing it.

Here's where a sticky contractual problem arises. For you to have agreed to the terms of the license, you must have read them first, otherwise the transaction must be considered a sale of the software. Thus, if the license is contained inside the package, and the user doesn't see it until he or she gets home, there can be no license. You agreed to buy what you saw in the store, not what you saw once you got home. This is why companies are now placing their license agreements in a conspicuous place on the packaging so that they can argue that you should have read it, and therefore agreed to it. Some companies even require you to sign an agreement before you can have the software. All this is even more pertinent with the mail order business. You can hardly claim that a person who orders over the phone has read your license agreement. And placing your license terms in an ad is next to absurd.

As you can see, licensing is not all that attractive. If you put a bunch of fine print legalese on your packaging or in your ad, you run the substantial risk that the purchaser will be turned off and won't buy it. If you try instead to slip it inside your packaging to get a "license by intimidation" it simply will not work. Technically, the user owns the copy of the program and is not bound by all that fine print.

Thus, many companies choose to just sell copies. You have to make your own choice when you decide to market your own software.

On the other hand, you can choose to have someone else market your software. This can be done by licensing your software for royalties or by assigning all your rights for the software for a lump sum. Next time I will discuss the legal considerations involved in licensing or assigning rights in your software, and I'll discuss many of the standard contract provisions and things you should look for.

See you at *RAINBOWfest!*

This column is meant to be educational and informative. It is not intended as legal advice. If you should have a legal question you should consult an attorney of your choice.

The Laserworm and The Firefly

By Josef A. Laake

I had a strange idea for a video game one day and I'm just crazy enough to spend hours translating it into reality for all you fellow CoCo-nuts. Now you, too, can learn to think like a Laserworm! Just type in the following program and *RUN* it. It's pretty much self-explanatory, but here are a few pointers. First, you receive points for your shots as charted here:

Scoreboard

1st	shot----1,000	pts
2nd	shot------500	pts
3rd	shot------250	pts
4th	shot------100	pts
5th	shot------- 50	pts

In addition to shot points, after connecting five shots you receive bonus points for the amount of time left in your turn (as indicated by the remaining energy indicator line at the top of the game screen). Every five connected shots increases the level of play difficulty (as indicated in the upper right-hand corner). This number also acts as a bonus multiplier, so you can see that as difficulty of play increases you are rewarded for quick shots. You are also rewarded with an additional laserworm for every 10,000 points. (You have three laserworms to start with and the number of laserworms left is indicated by the laserworm cocoons in the upper left-hand corner.)

I have used the high speed *POKE* in this program, so if your CoCo can't handle high speed be sure to delete the following statements:

In line 26 *DELete* 'POKE65495,0'
In line 163 *DELete* 'POKE65494,0'
DELete line 200

There are several ways to lose your laserworms, but you'll find out what they are soon enough.

So grab your joystick (right joystick, that is) and go get those pesky fireflies!

```
6.....00DD 108.... 0E80
22.....0443 151....11DD
41..... 07A2 177.... 14E6
58..... 0A55 200.....1723
79..... 0C1E END... 19F8
```

The listing:

```
Ø ' LASERWORM & FIREFLY 6.3
1 ' 16K STRIPPED VERSION
2 ' (C) 1983 JOSEF A. LAAKE
3 GOTO18
```

```
4 LINE(M,R)-(F,E),PRESET:LINE(M,
R)-(F,E),PSET:PLAY"O5L255BA#G#FC
#O-G#DO-G":RETURN
5 Y=PEEK(65280):IFY=126ORY=254TH
EN108ELSERETURN
6 PLAY"CDEFGABO+CP8":RETURN
7 CLSRND(7)+1:FORA=0TO31:B=RND(9
)-1:SET(A,0,B):SET(0,A,B):SET(63
-A,0,B):SET(63,A,B):SET(A,31,B):
SET(63-A,31,B):NEXT:GOSUB6:RETUR
N
8 DRAW"BM45,3"+A$(INT(SC(3)/10))
+"BM53,3"+A$(SC(3)-INT(SC(3)/10)
*10)+"BM61,3"+A$(INT(SC(2)/10))
9 DRAW"BM69,3"+A$(SC(2)-INT(SC(2
)/10)*10)+"BM77,3"+A$(INT(SC(1)/
10))+"BM85,3"+A$(SC(1)-INT(SC(1)
/10)*10)
10 DRAW"BM239,3"+A$(INT((INT(H/5
))/10))+"BM247,3"+A$((INT(H/5))-
((INT((INT(H/5))/10))*10)):LINE(
100,5)-(235,5),PSET
11 ONG GOTO12,13,14,15,16,17
12 PUT(35,3)-(41,7),L,PSET
13 PUT(27,3)-(33,7),L,PSET
14 PUT(19,3)-(25,7),L,PSET
15 PUT(11,3)-(17,7),L,PSET
16 PUT(3,3)-(9,7),L,PSET
17 RETURN
18 PLAY"L12802":GOSUB7
19 PRINT@198," RESET HIGH SCORE
? ";:PRINT@295," FOR YES, HIT <Y
> ";:PRINT@327," FOR NO,  HIT <N
> ";:PRINT@358," OR JOYSTICK BUT
TON ";:PLAY"01"
20 A$=INKEY$:IFA$="Y"THEN21ELSEI
FA$="N"THEN22ELSEY=PEEK(65280):I
FY=126ORY=254THEN22ELSE20
21 FORY=16376TO16378:POKEY,0:NEX
T
22 GOSUB7:PRINT@106," CAT'S MEOW
 ";:GOSUB6:PRINT@135," ELECTROSO
FT INC. ";:GOSUB6:PRINT@171," PR
ESENTS ";:GOSUB6
23 PLAY"L1603GF#GEFGL8AL16GF#GL2
8CDEFGABO4CP8":PRINT@260," LASER
WORM ";:PLAY"O3C":PRINT@271," &
";:PLAY"G":PRINT@274," FIREFLY!
";:PLAY"O+CO1P8"
24 PRINT@358," COPYRIGHT (C) 198
3 ";:GOSUB6:PRINT@392," JOSEF A.
 LAAKE ";:GOSUB6:PLAY"L128":GOSU
B6
25 FORY=16379TO16383:POKEY,0:NEX
T:POKE16380,4
26 POKE65495.0:CLEAR350.16375
27 DIMK(8,5),L(7,5),Q(8,5),V(8,5
),A(0),B(0),C(0),D(0),E(0),F(0),
G(0),H(0),M(0),R(0),X(0),Y(0),HS
(3),SC(3),T$(8),A$(9)
28 G=PEEK(16380):H=PEEK(16379):S
C(1)=PEEK(16381):SC(2)=PEEK(1638
2):SC(3)=PEEK(16383)
29 SC=SC(1)+SC(2)*100+SC(3)*1000
0:HS(1)=PEEK(16376):HS(2)=PEEK(1
6377):HS(3)=PEEK(16378)
30 HS=HS(1)+HS(2)*100+HS(3)*1000
0:T$(1)="BL7R1NE3F3
31 T$(2)="BR7L1NH3G3
32 T$(3)="BU7D1NG3F3
33 T$(4)="BD7U1NH3E3
34 T$(5)="BE6G1ND3L4
35 T$(6)="BH6F1ND3R4
36 T$(7)="BG6E1NU3R4
37 T$(8)="BF6H1NU3L4
38 A$(0)="BR1R2F1D4G1L2H1U4
39 A$(1)="BF1E1D6L2R4
40 A$(2)="R3F1D1G1L2G1D2R4
41 A$(3)="BD1E1R3F1D1G1L1R1F1D1G
1L3H1
42 A$(4)="D4R4U2D4
43 A$(5)="BR4L4D2R3F1D2G1L3
44 A$(6)="BR3L2G1D4F1R2E1U1H1L3
45 A$(7)="D1U1R4D2G2D2
46 A$(8)="BR1R2F1D1G1F1D1G1L2H1U
```

```
1E1NR1H1U1
47 A$(9)="BF3L2H1U1E1R2F1D4G1L2
48 PMODE4,1:PCLS:CIRCLE(4,20),4,
5,.9:PAINT(4,20),5,5
49 DRAW"BM1,1F3R1E3D1G3L1H3D1F3R
1E3D1G3L1H3BM1,15E3R1F3U1H3L1G3U
1E3R1F3U1H3L1G3
50 GET(1,2)-(8,6),V,G:GET(1,10)-
(8,14),Q,G:GET(9,2)-(16,6),K,G:G
ET(1,18)-(7,22),L,G:M=122:R=183:
I=1:K=1
51 PMODE4,1:PCLS:SCREEN1,1:GOSUB
8:TIMER=0
52 A=RND(247):B=RND(172)+11
53 PUT(A,B+3)-(A+7,B+7),K,PSET
54 A=A+RND(INT(((2*H)+45)/5))-IN
T((H+25)/5):B=B+RND(INT(((2*H)+4
5)/5))-INT((H+25)/5):IFA<0THENA=
0
55 IFB<11THENB=11
56 IFA>248THENA=248
57 IFB>182THENB=182
58 PUT(A,B)-(A+7,B+4),V,PSET:GOS
UB62:GOSUB5:PUT(A,B)-(A+7,B+7),K
,PSET:PUT(A,B+3)-(A+7,B+7),Q,PSE
T
59 IFTIMER>1240THENB=3:GOTO159
60 Y=TIMER/9+100:LINE(100,5)-(Y,
5),PRESET
61 IFM<A ORM>A+7 ORR<B ORR>B+7TH
EN53ELSEB=1:GOTO159
62 C=JOYSTK(0):D=JOYSTK(1):POKE2
00,M:POKE202,R:DRAW"C0"+T$(K):PL
AY"L6403G"
63 IFC=63 AND D=63THEN80
64 IFC=0 AND D=63THEN83
65 IFC=0 AND D=0THEN86
66 IFC=63 AND D=0THEN89
67 IFD=63THEN92
68 IFD=0THEN96
69 IFC=63THEN100
70 IFC=0THEN104
71 PUT(M-3,R-2)-(M+3,R+2),L,PSET
:POKE200,M:POKE202,R:DRAW"C5"+T$
(K):ONK GOTO72,73,74,75,76,77,78
,79
72 PLAY"1":RETURN
73 PLAY"8":RETURN
74 PLAY"5":RETURN
75 PLAY"12":RETURN
76 PLAY"6":RETURN
77 PLAY"3":RETURN
78 PLAY"0+1":RETURN
79 PLAY"10":RETURN
80 M=M+4:IFM>247THENM=247:GOTO94
81 R=R+4:IFR>183THENR=183:M=M-4:
GOTO102
82 K=8:GOTO71
```

```
83 M=M-4:IFM<10THENM=10:GOTO94
84 R=R+4:IFR>183THENR=183:M=M+4:
GOTO106
85 K=7:GOTO71
86 M=M-4:IFM<10THENM=10:GOTO98
87 R=R-4:IFR<16THENR=16:M=M+4:GO
TO106
88 K=6:GOTO71
89 M=M+4:IFM>247THENM=247:GOTO98
90 R=R-4:IFR<16THENR=16:M=M-4:GO
TO102
91 K=5:GOTO71
92 IFM<10THENM=10
93 IFM>247THENM=247
94 R=R+5:IFR>183THENR=183
95 K=4:GOTO71
96 IFM<10THENM=10
97 IFM>247THENM=247
98 R=R-5:IFR<16THENR=16
99 K=3:GOTO71
100 IFR>183THENR=183
101 IFR<16THENR=16
102 M=M+7:IFM>247THENM=247
103 K=2:GOTO71
104 IFR>183THENR=183
105 IFR<16THENR=16
106 M=M-7:IFM<10THENM=10
107 K=1:GOTO71
108 X=TIMER:ONK GOTO109,112,115,
118,121,126,131,136
109 E=R:FORF=M-8TO2STEP-2
110 IFPPOINT(F,E)=5THEN181
111 PSET(F,E):NEXTF:GOTO141
112 E=R:FORF=M+8TO254STEP2
113 IFPPOINT(F,E)=5THEN181
114 PSET(F,E):NEXTF:GOTO141
115 F=M:FORE=R-8TO11STEP-2
116 IFPPOINT(F,E)=5THEN181
117 PSET(F,E):NEXTE:GOTO141
118 F=M:FORE=R+8TO191STEP2
119 IFPPOINT(F,E)=5THEN181
120 PSET(F,E):NEXTE:GOTO141
121 E=R-7:FORF=M+7TO255STEP2
122 IFPPOINT(F,E)=5THEN181
123 PSET(F,E):E=E-2
124 IFE<10THENE=E+2:GOTO141
125 NEXTF:F=F-2:E=E+2:GOTO141
126 F=M-7:FORE=R-7TO11STEP-2
127 IFPPOINT(F,E)=5THEN181
128 PSET(F,E):F=F-2
129 IFF<1THENF=F+2:GOTO141
130 NEXTE:GOTO141
131 F=M-7:FORE=R+7TO191STEP2
132 IFPPOINT(F,E)=5THEN181
133 PSET(F,E):F=F-2
134 IFF<0THENF=F+2:GOTO141
135 NEXTE:E=E-2:F=F+2:GOTO141
136 F=M+7:FORE=R+7TO191STEP2
```

```
137 IFPPOINT(F,E)=5THEN181
138 PSET(F,E):F=F+2
139 IFF>255THENF=F-2:GOTO141
140 NEXTE:E=E-2:F=F-2:GOTO141
141 GOSUB4
142 LINE(M,R)-(F,E),PRESET
143 I=I+1:IFI=6THENB=2:GOTO159
144 TIMER=X:RETURN
145 PLAY"L6402":GOSUB7
146 PRINT@33," FINAL  SCORE    ";
147 PRINT@49,"   HIGH SCORE ";
148 IFHS<SC THENHS=SC ELSE152
149 POKE16376,SC(1)
150 POKE16377,SC(2)
151 POKE16378,SC(3)
152 PRINT@67,SC;:PRINT@86,HS;
153 PRINT@233," *** GAME *** ";
154 PRINT@265," *** OVER *** ";
155 PRINT@417," TO RESET GAME,
PRESS <R> OR ";
156 PRINT@450," FIRE  BUTTON  ON
  JOYSTICK ";
157 Y=PEEK(65280):IFY=126 OR Y=2
54THEN3
158 IFINKEY$="R"THEN3ELSE157
159 PLAY"L28":FORY=5TO55STEP5
160 CIRCLE(M,R),Y,,,.9
161 PLAY"L-L-01;"+STR$(13-(Y/5))
162 NEXTY:G=G+1:POKE16380,G
163 POKE65494,0:CLS5
164 ONB GOTO165,176,179,180
165 PRINT@68," FIREFLY IN YOUR E
YE!!! ";
166 PRINT@132," FATAL TO LASERWO
RMS!!! ";
167 PLAY"L801CDE-FE-P8CP8L44G#AG
#AG#AG#AG#AL8GP16L16B02CP1601GP1
6CP2"
168 IFG=6THEN170ELSEIFG=7THEN145
169 PRINT@258," YOU HAVE   LASER
WORMS LEFT ";:GOTO171
170 PRINT@258," YOU HAVE   LASER
WORM LEFT! ";
171 FORY=1TO5:PRINT@268,CHR$(128
);
172 PLAY"L12601CO2CO1CO2CO1CO2C"
173 PRINT@267,7-G;
174 PLAY"O1GO2GO1GO2GO1GO2G"
175 NEXT:GOTO26
176 PRINT@71," TOO MANY SHOTS!!
";
177 PRINT@132," DEPLETED ENERGY
SUPPLY ";
178 PRINT@195," IS FATAL TO LASE
RWORMS!! ";:GOTO167
179 PRINT@72," OUT OF TIME!!! ";
:GOTO177
180 PRINT@69," YOU SHOT YOURSELF
!!! ";:GOTO166
181 GOSUB4
182 IFF<A OR F>A+7 OR E<B OR E>B
+7THENB=4:GOTO159
183 PLAY"L80O4":FORY=15TO39STEP3
184 CIRCLE(F,E),Y-10,,,.9
185 IFY<39THENPLAYSTR$(Y/3):NEXT
ELSEPLAY"O5C":C=0:H=H+1
186 ONI GOTO187,188,189,190,191
187 C=500
188 C=C+250
189 C=C+150
190 C=C+50
191 C=C+50
192 SC=SC+C:B=INT((1240-X)*(INT(
H/5)/24.8))
193 IFX>930ORC=50THENB=0ELSESC=S
C+B
194 SC(3)=INT(SC/10000)
195 SC(2)=INT((SC-SC(3)*10000)/1
00)
196 SC(1)=SC-SC(3)*10000-SC(2)*1
00
197 POKE16380,G:POKE16379,H
198 POKE16381,SC(1)
199 POKE16382,SC(2)
200 POKE 65494,0
201 CLS5:PRINT@228," POINTS THIS
  SHOT "C;
202 IFH<5THEN204ELSEPRINT@260,"
ENERGY BONUS  X"INT(H/5);B;
203 PRINT@292," TOTAL  THIS  ROU
ND"B+C;
204 ONI GOTO205,207,206,208,209
205 PLAY"L3204CDEO+CEGECO-ECEGO+
C":GOTO210
206 PLAY"L1604EDCO-AO+AP16GO+C":
GOTO210
207 PLAY"L2004GFGDGO-BO+CDO-GO+C
":GOTO210
208 PLAY"L1603EE-EE-EDCO-AGO+C":
GOTO210
209 PLAY"L6003E-EP10G-GP30E-EP6E
-EP10G-GP30E-E"
210 IFSC(3)>PEEK(16383)THENPOKE1
6383,SC(3):G=G-1ELSE26
211 POKE16380,G
212 CLS5:PRINT@70," CONGRADULATI
ONS!!! ";
213 PRINT@130," EVERY 10,000 ENE
RGY POINTS ";
214 PRINT@199," HATCHES ANOTHER
";
215 PLAY"L1603GEGO4CO3GECDEDO2GO
3L48CDCECFCGCACBCO4L16CP1605CP6"
216 FORX=1TO5:PRINT@262,STRING$(
20,143+X*16);:PLAY"L6405GCGCGC"
217 PRINT@262," LASERWORM LARVA!
!! ";:PLAY"GB-GB-GB-GB-":NEXT
218 PRINT@322," YOU NOW HAVE"7-G
"LASERWORMS! ";:PLAY"P1":GOTO26
```

RAINBOW

Give us your best: Join the ranks of these courageous CoCoists in showing the Color Computer world your high score at your favorite micro-diversion. We want to put your best effort on record in *the Rainbow's* Scoreboard column. All entries must be received by the first of the month to be eligible for the following month's Scoreboard. Each individual is limited to three score entries per month.

★ **New Number One** • **Last Month's Number One**

ALCATRAZ II *(Spectral Associates)*
18,790 ★Chris Sweet, Harvard, MA
8,710 •Kanti Dinda, Kingston, Ontario

ANDROID ATTACK
25,000 ★Wes Hill, Vashon, WA
15,500 Cameron Amick, Reisterstown, MD

ASSAULT *(Computer Shack)*
1,100 ★Kevin Mesecher, Ft. Walton Beach, FL

ASTEROID
2,322 ★Matt McMann, New Boston, MI

ASTRO BLAST *(Mark Data)*
158,000 ★Larry Plaxton, Medley, Alberta
157,000 Scott Drake, Pine City, NY
104,464 Jim Baker, Florissant, MO
98,000 Tim Warr, Bellingham, WA
92,000 Harry Sawyer, Watchung, NJ
46,250 Brenda Teel, Martintown, Ontario

AVENGER *(The Cornsoft Group)*
24,575 ★Rich McGervey, Morgantown, WV
19,480 •Jim Sparke
19,220 Vince Lok, Mississauga, Ontario
16,310 Dave Lubnow, Sussex, NJ
14,075 Stephen Lai, Palatine, IL

BIRD ATTACK *(Tom Mix)*
200,725 ★Peter Niessen, Carlisle, MA
110,850 Chris Sweet, Harvard, MA

BLEEP
105 ★Matt McMann, New Boston, MI

BLOC HEAD *(Computerware)*
286,900 ★Ron Moore, Greensburg, PA
153,825 Greg Redden, Halifax, NS
124,925 Mike Rausch, Denver, CO
113,225 Beth Walker, Gloucester Pt., VA
107,500 Pat Davis, Bothell, WA
63,475 Don Massey, Bothell, WA
61,475 Hal Mermelstein, S. Windham, ME
47,925 David Plesser, El Cajon, CA

BUSTOUT *(Radio Shack)*
42,000 ★Derrick Kardos, Colonia, NJ
34,700 Sara Hennessey, Golden Valley, MN
28,720 Perry Denton, New Baden, IL
27,880 Mike Wells, Pittsburgh, PA
17,170 Neil Berkman, DeWitt, NJ

BUZZARD BAIT *(Tom Mix)*
116,450 ★Steve Sentell, Seymour, TN
102,450 Steve Sytsma, Big Rapids, MI

CANYON CLIMBER *(Radio Shack)*
155,900 ★Dustin Yates, Shreveport, LA
144,000 Andy Herron, High Point, NC
142,100 Jim Corkrey, Leesburg, VA
140,000 Peter Johnson, Chino, CA
133,400 Kevin Dawdy, Findlay, IL
126,400 Kristin Vitek, Hillside, IL
113,000 Gary Smith, North Bay, Ontario
111,800 Philippe Moisan, St. Jerome, Quebec
99,800 Joe Nasal, Doylestown, PA
64,000 John Mattox, Chattanooga, TN
63,000 Matthew S. Kerr, Plainfield, NJ
61,400 Abdullah VanDer Breggen, Regina, Saskatchewan

CATCH 'EM *(Aardvark)*
237,000 ★Craig Edelheit, W. Bloomfield, MI
91,000 Dean Bouchard, Kingston, N.S.
70,157 Kirk Beler, Taber, Alberta
65,768 Laura Sandman, Louisville, KY

CATERPILLAR *(Aardvark)*
180,627 ★Brian Panepinto, Spencerport, NY
63,100 •Todd Byington, N. Salt Lake, UT
54,608 Lawrence McElligott, Lanc ster, CA
53,233 Ron Pinson, Guelph, Ontario
46,641 Laura Schooley, Richmond, VA
44,853 Felicia Schooley, Richmond, VA
42,563 Doug Schofield, Plainfield, IL
15,029 Michael Jimenez, Mesa, AZ
10,148 Steven Jimenez, Mesa, AZ

CAVE HUNTER *(Mark Data)*
42,600 ★Gary Ritchie, Bellevue, Alberta
27,050 Jim Baker, Florissant, MO
26,300 Mike Hughey, King George, VA
21,150 Jim Baker, Florissant, MO
15,200 Rick McGerney, Morgantown, WV

CLOWNS & BALLOONS *(Radio Shack)*
85,660 ★Teresa Stutsman, N. Little Rock, AR
83,710 •Don Fraser, Shakope, MN
77,910 Dan James, Clearwater, FL
74,920 Sal Bartett, Mesa, AZ
74,600 Jimmy Kendall, Elkview, WV
30,030 Kevin Schofield, Plainfield, IL

COLOR CUBES (Radio Shack)
4:50 ★John Handis
6:05 Chris Cope, Central, SC

COLOR HAYWIRE *(Mark Data)*
22,050 ★Jim Baker, Florissant, MO
17,850 Brian Wallingford, Fall River, MA
14,850 Michael Rhattigan, Cary, NC
14,750 Rich McGervey, Morgantown, WV
12,450 Chris Coyle, Selden, NY

COLOR INVADERS *(Computerware)*
240,700 ★Roland Hendel, Mississauga, Ontario
227,050 John Osborne, Kincardine, Ontario
217,635 Jim Baker, Florissant, MO
126,350 Perry Denton, New Baden, IL
93,510 Rich McGervey, Morgantown, WV

COLOR METEROIDS *(Spectral Associates)*
1,496,000 ★Craig Edelheit, W. Bloomfield, MI
1,253,200 Jeff White, Prairie de Chien, WI
292,000 Roland Hendel, Mississauga, Ontario
292,000 Vince Lok, Mississauga, Ontario
252,050 Larry Plaxton, Medley, Ontario

COLOR OUTHOUSE *(Computer Shack)*
35,908 ★Ron Rhead, Ontario, Canada

COLORPEDE *(Intracolor)*
3,355,248 ★Scott Drake, Pine City, NY
2,745,982 •Roland Hendel, Mississauga, Ontario
2,547,299 Rich McGervey, Morgantown, WV
2,400,735 Scott Sehlhorst, Columbia, SC
2,139,248 Mike Hall, Hartland, WI
1,102,724 Bryan Andrews, Plainfield, IN
339,438 David Gibson, Longwood, FL
207,249 Margaret Dunlop, Trail, BC
173,564 Kenny Miller, Yardley, PA

COLOR SCARFMAN *(The Cornsoft Group)*
976,520 ★Bruce Thornhill, Barrhead, Alberta
772,000 Keith Seifried, Greenville, OH
600,410 Roland Hendel, Mississauga, Ontario
539,100 Larry Plaxton, Medley, Alberta
530,200 Alan Klotzback, Plainesville, OH
300,690 Frankie Jimenez, Mesa, AZ
222,000 Steven Jimenez, Mesa, AZ
110,690 Michael Jimenez, Mesa, AZ

COLOR ZAP *(Spectral Associates)*
227,330 ★Ron Rhead, Willowdale, Ontario
84,400 Chris Sweet, Harvard, MA
82,710 Neil Berkman, DeWitt, NY
50,800 Scott Sehlhorst, Columbia, SC

COLOUR PAC ATTACK *(Computerware)*
472,465 ★Jim Baker, Florissant, MO
211,000 Roland Hendel, Mississauga, Ontario
193,000 Cameron Amick, Reisterstown, MD
189,540 Mark Nichols, Birsay, Saskatchewan
154,600 Greg Lesher, North Chili, NY

CONQUEST OF KZIRGLA *(Rainbow Connection Software)*
50,199 ★Bruce Uher, Coshocton, OH
10,399 Scott Sehlhorst, Columbia, SC

COSMIC CLONES *(Mark Data)*
41,300 ★John Osborne, Kincardine, Ontario

CROSSWORDS *(Radio Shack)*
864 ★Bob Strang, Chicago, IL

DEATH TRAP *(Soft Sector)*
89,035 ★Keith Philabaum, Coschocton, OH
84,672 •Sandra Willard
75,431 Jeff Willard, Chireno, TX
41,490 M. A. Brickler, Allen Park, MI
41,272 Perry Willard, Chireno, TX

DEFENSE *(Spectral Associates)*
68,750 ★M. A. Brickler, Allen Park, MI
58,900 •Greg Scott, Orlando, FL

DOODLEBUG *(Computerware)*
2,248,840 ★Donut, Tabor, IA
1,580,000 J.W. Panks, Indianapolis, IN
1,577,320 •Bobby Bruce, Springfield, MO
1,265,580 Lawrence McElligott, Lancaster, CA
1,152,360 Marc Hassler
747,180 Scott Snyder, Allentown, PA
518,820 Greg Redden, Halifax, NS
286,560 Kyle Keller, Overland Park, KS

DOUBLEBACK *(Radio Shack)*
1,080,000 ★Phillipe Duplanties, St. Jerome, Quebec
474,040 Paul Moritz, Butte, MT
435,570 •Phillippe Morsan, St. Jerome, Quebec
429,000 Steve Damm, Phoenix, AZ
378,750 Jim Wolf, South Bend, IN
127,460 Kyle Keller, Overland Park, KS
119,390 Johanne Duplanties, St. Jerome, Quebec
115,840 France Duplanties, St. Jerome, Quebec

DUNKEY MUNKEY *(Intellectronics)*
3,214,200 ★Rich McGervey, Morgantown, WV
1,924,000 Andrew Herron, High Point, NC
1,618,800 Bryan Bloodworth, Federal Way, WA
1,437,200 Mike Wells, Pittsburgh, PA
1,245,000 Jodi Knudsen, Kenosha, WI
778,400 Michael McCafferty, Idaho Falls, ID
603,100 Bryan Teel, Martintown, Ontario

FAST LANE *(Ace Soft Computer Products)*
23,782 ★Philip Deen, Enterprise, FL
93 Marie Love, Columbia, SC

FIRECOPTER *(Adventure International)*
113,880 ★Steve Skrzyniarz, Tacoma, WA
74,640 Cameron Amick, Reisterstown, MD

FLYBY
3,670 ★Cameron Amick, Reisterstown, MD

THE FROG *(Tom Mix)*
73,350 ★Evelyn Gagnon, Ontario, Canada
66,480 Jeannie Hawkins, Deltona, FL
47,360 •Jeannie Hawkins, Deltona, FL
36,260 Chris Coyle, Selden, NY
31,480 Aimee Bergeron, Carlisle, MA
24,700 David Gibson, Longwood, FL
14,540 Damon Frazier, Bringhurst, IN

FROGGER *(The Cornsoft Group)*
32,010 ★Laura Schooley, Richmond, VA
25,425 •Kanti Dinda, Kingston, Ontario
20,745 Felicia Schooley, Richmond, VA
18,575 Tamara Schooley, Richmond, VA
17,945 Steve Skrzyniarz, Tacoma, WA
15,050 Steve Hemphill, Los Angeles, CA

FROG-MAN *(Computer Island)*
3,735 ★Aaron T. Cincotta, Peru, IN

FROG TREK *(Oelrich Publications)*
14,700 ★Greg Burke, Kenora, Ontario
14,060 Alan Weiss, Summit, NJ
11,200 Jim Partridge, Clinton, CT
10,370 Jim Baker, Florissant, MO
8,550 Tom Eberhardt, Fair iew Park, OH

SCOREBOARD

FURY *(Computer Shack)*
54,300 ★Hans Haimberger, Freewater, Ontario

GALACTIC ATTACK *(Radio Shack)*
67,750 ★Chuck Gaudette, Monroe, CT
58,000 Terry Steen, San Bernadino, CA
55,360 Donald Thompson, Lubbock, TX
54,200 Mike Hughey, King George, VA
54,000 Craig Edelheit, W. Bloomfield, MI
23,670 Laura Schooley, Richmond, VA
20,250 Felicia Schooley, Richmond, VA
16,260 Joan Haysom, Ottawa, Ontario

GALAX ATTAX *(Spectral Associates)*
62,650 ★Steve Hargis, Tucson, AZ
73,000 Wes Hill, Vashon, WA
66,750 Jim Wolf, South Bend, IN
65,700 Steve Otis, Graham, WA
60,950 ●Dennis Goehring, Mississauga, Ontario

GHOST GOBBLER *(Spectral Associates)*
1,007,430 ★Todd Brannam, Charleston Hts., SC
825,250 ●Randy Gerber, Wilmette, IL
423,390 Rich McGervey, Morgantown, WV
255,000 John Osborne, Kincardine, Ontario
142,070 Chris Kulawy, St. Johnsville, NY
92,350 Geno Benick, Rice, MN
76,590 Steve Curnow, Woodbury, MN
72,250 Greg Redden, Halifax, NS
48,300 Chris Leonard, Minoa, NY

GOLF *(Aardvark)*
40 ★Matthew Brenengen, Lakelmo, MN

GRABBER *(Tom Mix)*
50,250 ★Doug Rodger, Harvard, MA
44,100 ●Steve Skrzyniarz, Tacoma, WA
31,200 Cameron Amick, Reisterstown, MD
30,600 Phil Calandrino, Springfield, IL

GRANPRIX *(Computerware)*
5,875 ★Steve Skrzyniarz, Tacoma, WA

HAIDER KHAZEN
81,000 ★David Karam, Austin, TX

HALL *(Chromasette)*
3,650 ★Michael McCafferty, Idaho Falls, ID

INVADERS REVENGE *(Med Systems)*
451,060 ★John Osborne, Kincardine, Ontario
32,600 Harry Sawyer, Watchung, NJ
16,300 Eric Lund, Millington, NJ

JUNIOR'S REVENGE *(Computerware)*
1,115,300 ★Ryan Van Manen, Grand Rapids, MI
144,200 Rich Van Manen, Grand Rapids, MI

KATERPILLAR ATTACK *(Tom Mix)*
163,526 ★Andy Truesdale, Ferguson, MO
109,247 Jamie Gritton, Irvine, CA
14,375 Norbert Berenyi, Northvale, NJ
14,211 Roland Hendel, Mississauga, Ontario
12,703 Warren Schubert, Newfoundland, NJ
8,504 Chris Leonard, Minoa, NY

KEYS OF THE WIZARD *(Spectral Associates)*
640 ★Steve Skrzyniarz, Tacoma, WA

THE KING *(Tom Mix)*
4,040,300 ★Andy Truesdale, Ferguson, MO
2,410,200 Candy Harden, Birmingham, AL
2,213,000 James Quadarella, Brooklyn, NY
1,858,000 Mike Hughey, King George, VA
1,670,200 Joel Doucet, Yarmouth, Nova Scotia
604,500 Curtis Giles, Altamonte Springs, FL
469,600 Kyle Keller, Overland Park, KS
289,400 Brian Panepinto, Spencerport, NY
232,600 Rich Trawick, N. Adams, MI
175,300 Brian Chaples, Springfield, VA
167,720 Scott Snyder, Allentown, PA
142,700 Michael Rosenberg, Prestonsburg, KY
127,000 Kenny Miller, Yardley, PA
91,600 Bryan Teel, Martintown, Ontario
82,700 Frankie Jimenez, Mesa, AZ
64,000 Chris Leonard, Minda, NY

KOSMIC KAMIKAZE *(IMB)*
200,550 ★Fred Iha, Columbus AFB, MS
62,250 Beth Walker, Gloucester Pt., VA
55,000 Paul Morris, Richland, WA
49,900 Mark Raphael, Englishtown, NJ
30,450 Jeff Teague, Noblesville, IN

LANCER *(Spectral Associates)*
474,250 ★Mike Rausch, Denver, CO
469,400 Jeff Jackson, Littleton, CO
462,100 Scott Jackson, Littleton, CO
183,050 Larry Sandhaas, Springfield, IL
134,900 Grant Gillott, Calgary, Alberta
112,250 Brian Chaples, Springfield, VA
105,750 Alex Barr, Littleton, CO
93,750 Kurt Kinney, Whitesboro, NY
84,250 Michael Rhattigan, Cary, NC
74,050 Nick Lewis, Cumming, GA
66,600 Gary Davis, Cary, NC
53,700 Daniel Shat, Chalfont, PA

MEGA-BUG *(Radio Shack)*
60,000 ★Robin Worthem, Milwaukee, WI
16,632 John Tiffany, Washington, D.C.
15,999 Ed Mitchell, Ragged Mountain, CO
14,297 Aleisha Hemphill, Los Angeles, CA
13,852 Ryan Van Manen, Grand Rapids, MI
5,709 Joan Haysom, Cttawa, Ontario

MEGAPEDE *(Computerware)*
89,038 ★Mark Skala, Fairview, PA
67,605 ●Ed Bottini, St. Louis, MO
58,070 Rich Van Manen, Grand Rapids, MI
55,291 Ryan Van Manen, Grand Rapids, MI
53,019 Jim Ganninger, Des Peres MO

METEOROIDS *(Spectral Associates)*
140,210 ★Fred Iha, Columbus AFB, MS

METEORS
403,000 ★Jimmy Chan, Kincardine, Ontario
17,810 ●Lenny Munitz, Bellerose, NY
13,120 Jennifer Klamp, Winter Park, FL

MICROBES *(Radio Shack)*
318,830 ★Hwan Joo, Weston, Ontario
316,200 Steve Mayer & Keith Seifried, Greenville, OH
259,700 Sheila Coleman, Griffin, GA
258,150 Rick Van Manen, Grand Rapids, MI
244,700 Ian Waters, North Hero, VT

MR. MUNCH
40,000 ★Wes Hill, Vashon, WA
20,000 Scott Drake, Pine City, NY

MONKEY KONG *(Med Systems Software)*
1,000,000 ★Wes Hill, Vashon, WA
1,876 Kanti Dinda, Kingston, Ontario
1,323 Ryan Van Manen, Grand Rapids, MI
1,210 Susan Brink, Portage, MI
1,185 Mike Fath, Wadsworth, OH
1,106 Kelly Kerr, Wentzville, MO

MONSTER MAZE *(Radio Shack)*
520,470 ★Robert L. Bull, Trenton, Ontario
323,490 Philip Morrissey, Gilboa, NY
200,000 Daniel Milbrath, Ann Arbor, MI
150,000 Jamie Gritton, Irvine, CA
114,000 Michael Daley, Biloxi, MS

MOON HOPPER *(Computerware)*
809,750 ★Ed Bottini, St. Louis, MO
142,800 Ron Rhead, Willowdale, Ontario
100,640 Cameron Amick, Reisterstown, MD

MOROCCO GRAN PRIX *(Computerware)*
4,153 ★Frank Bottini, St. Louis, MO
2,344 Cameron Amick, Reisterstown, MD

NERBLE FORCE *(Computerware)*
315,150 ★Frank Bottini, St. Louis, MO
49,200 Jim Ganninger, Des Peres, MO

NIBBLER *(Color Quest)*
20,650 ★Rick McGervey, Morgantown, WV
14,910 ●Christal Glovinsky, Staten Island, NY

NINJA WARRIOR *(The Programmer's Guild)*
74,500 ★Greg Burke, Kenora, Ontario
39,900 Dennis Teague, Noblesville, IN
29,700 Damon Frazier, Bringhurst, IN
27,500 ●Jeff Teague, Noblesville, IN

OFFENDER *(American Business Computers)*
999,900 ★Scott Drake, Pine City, NY
965,400 ●Jim Baker, Florissant, MO
406,800 Betty Moore, Greensburg, PA
293,900 Marla Moore, Greensburg, PA
273,900 Walt Moore, Greensburg, PA

PAC-ATTACK II *(Computerware)*
107,070 ★Mark Skala, Fairview, PA
88,860 Michael J. Garozzo, Morrisville, PA
62,870 ●Stanley Sneed, Erwin, TN
44,000 Robert Lang, Port McNeill, BC
42,045 Dave Lubnow, Sussex, NJ
33,900 Jean Archambault, Quebec, Canada
30,995 Bryan Teel, Martintown, Ontario
30,015 Joan Haysom, Ottawa, Ontario

PAC-DROIDS *(The Programmer's Guild)*
1,476,730 ★Joshua Josephson, Corning, NY
577,140 ●Richard Cochrane, Wayne, NJ
151,590 Annita Powell, Huber Heights, OH
140,300 John Yapp, Park Forest, IL
94,140 Lee Powell, Huber Heights, OH

PACET-MAN *(American Business Computers)*
26,600 ★Vince Lok, Mississauga, Ontario
14,050 Susan Brink, Portage, MI
7,200 Fred Iha, Columbus AFB, MS
5,000 Cameron Amick, Reisterstown, MD
3,392 Norbert Berenyi, Northvale, NJ

PARACHUTE JUMP *(JARB Software)*
451,000 ★John Osborne, Kincardine, Ontario

PHANTOM SLAYER *(Med Systems)*
1,306 ★Marc Hassler
604 J. Powell, Bournemouth, England
406 Imre Kertesz, Chandler, AZ
197 Mark Heizler, Ridgefield, CT
160 Mike Hall, Hartland, WI

PICNIC *(Computer Island)*
100,000 ★Scott Drake, Pine City, NY
1,220 Jon Bauch, South Fallsburgh, NY

PINBALL *(Radio Shack)*
4,000,000 ★Keith Seifried, Greenville, OH
2,111,900 Dale Westmoreland, Lannon, WI
1,266,250 Donut, Tabor, IA
300,250 Brad Widdup, Dundas, Ontario
258,600 Michael Rosenberg, Prestonsburg, KY

PLANET INVASION *(Spectral Associates)*
483,250 ★Chris Sweet, Harvard, MA
286,075 ●Larry Plaxton, Medley, Alberta
257,900 Ron Rhead, Willowdale, Ontario
221,350 John Cole, King City, Ontario
207,150 Mike Hughey, King George, VA

PLANET RAIDERS *(Aardvark)*
2,010,900 ★Bill Messerick, S. St. Paul, MN

POLARIS *(Radio Shack)*
256,018 ★Michael Popovich, Sr., Nashua, NH
218,450 Allen Roth, Dayton, TX
212,746 Hwan Joo, Weston, Ontario
170,100 Steve Johnson, Santa Ana, CA
151,154 Brian Austin, Rotterdam, NY
39,746 Steve Curnow, Woodbury, MN

POLTERGEIST *(Radio Shack)*
6,455 ★Rick Van Manen, Grand Rapids, MI
5,021 Rich Van Manen, Grand Rapids, MI
4,978 ●Tim Warr, Bellingham, WA
4,956 Mark Dowling, San Bruno, CA
4,835 Scott Butler, Springfield, IL
4,710 Don Turowski, Natrona Heights, PA
4,640 Joe Nasal, Doylestown, PA
4,455 Scott Snyder, Allentown, PA
4,395 Joan Haysom, Ottawa, Ontario

POPCORN *(Radio Shack)*
560,900 ★Vince Lok, Mississauga, Ontario
168,680 Steve Johnson, Santa Ana, CA
136,530 Scott Sehlhorst, Columbia, SC
110,570 Cameron Amick, Reistertown, MD
96,470 Mike Wells, Pittsburgh, PA
38,360 Michael Rosenberg, Prestonsburg, KY

continued

MORE... RAINBOW SCOREBOARD

PROJECT NEBULA
430 ★Dale Westmoreland, Lannon, WI

PROTECTORS *(Tom Mix)*
594,610 ★Roland Hendel, Mississauga, Ontario
358,514 Cameron Amick, Reisterstown, MD
275,810 Julian Bond, Berkeley, CA
272,000 Douglas Hug, Roseville, CA
217,000 Ian Waters, North Hero, VT
106,750 David Gibson, Longwood, FL
43,650 Mike Hishan, London, Ontario
40,500 Michael Jimenez, Mesa, AZ

PYRAMID *(Radio Shack)*
165 ★Dale Westmoreland, Lannon, WI

RAIL RUNNER *(Computerware)*
53,520 ★Jim Baker, Florissant, MO
53,400 Ed Bottini, St. Louis, MO
19,400 Chris Sweet, Harvard, MA
15,520 Kirk Mesecher, Ft. Walton Beach, FL
12,275 Larry Mescher, Ft. Walton Beach, FL

ROBOT BATTLE *(Spectral Associates)*
21,000 ★Bobby Bruce, Springfield, MO

ROBOTTACK *(Intracolor)*
1,512,200 ★Robert Kiser, Monticello, MS
1,219,810 ●Steve Skrzyniarz, Tacoma, WA
1,197,800 Brian Austin, Rotterdam, Ny
1,146,750 Justin Marcus
1,107,400 John Osborne, Kincardine, Ontario
807,700 Jeff Jackson, Littleton, CO
559,750 Philippe Moisan, St. Jerome, Quebec

SANDS OF EGYPT *(Radio Shack)*
121 ★Grant Gillot, Calgary, Alberta
88 Mike Rausch, Denver, CO

SCARFMAN *(Cornsoft)*
495,440 ★Woody Farmer, Alberta, Canada
371,540 ●Stanley Sn ed, Erwin, TN
121,600 M. A. Brickles, Allen Park, MI

SEA DRAGON *(Adventure International)*
60,430 ★Steve Skrzyniarz, Tacoma, WA
36,540 Steve Schweitzer, Sewell, NJ
27,500 Wes Hill, Vashon, WA
23,750 Stephen Har en, Birmingham, AL
15,300 Derek Keener, Columbus, OH

SHAFT *(Prickly-Pear Software)*
18,150 ★Loren Seng, Tuscon, AZ
17,160 Julian Bond, Berkely, CA

SHARK TREASURE *(Computerware)*
72,000 ★Marc Hassler
60,000 Aaron Peelle, Bennington, VA
50,000 Jennifer Klamp, Winter Park, FL
46,000 Ed Bottini, St. Louis, MO

SHOOTING GALLERY *(Radio Shack)*
37,300 ★Mark Nichols, Birsay, Saskatchewan
37,200 ●Chris Kulawy, St. Johnsville, NY,
37,080 Gary Long, Peru, NY
33,060 Mrs. Merle Burzynski, Erie, MI
31,230 D. A. Turowski, Natrona Heights, PA

SKIING *(Radio Shack)*
31.52 ★Andy Truesdale, Ferguson, MO
35.89 Peter Johnson, Chino, CA
40.10 Fred Herrmann, Flemington, NJ
49.43 John Scanlan, Prairie Village, KS

SKY DEFENSE *(Quasar Animation)*
9,700 ★Vince Lok, Mississauga, Ontario
6,700 Mike Anheluk, Fall Creek, OR
6,120 Steve Skrzyniarz, Tacoma, WA
5,200 Cameron Amick, Reisterstown, MD

SNAIL'S REVENGE *(the Rainbow)*
510 ★Matt Luck, McLean, VA
420 Christie James, Ashland, VA

SOLO POKER *(Radio Shack)*
640 ★Mary J. Herring, Bloomington, IL
640 ★Stephen Jenkick, Pittsburgh, PA

SOLO POOL *(Tom Mix)*
110 ★Bill Steelman, Edmonds, WA
103 ●John Fraysse, Dahlren, VA

SPACE ACE *(Spectral Associates)*
983 ★Matt McMann, New Boston, MI

SPACE AMBUSH *(Computerware)*
403,350 ★Jimmy Chan, Kincardine, Ontario
124,460 ●John Osborne, Kincardine, Ontario

SPACE ASSAULT *(Radio Shack)*
358,660 ★Mike Snelgrove, Oshawa, Ontario
238,560 ●John Cole, King City, Ontario
224,130 Derrick Kardos, Colonia, NJ
221,130 Steve Johnson,Santa Ana, CA
216,680 Chris Kulawy, St. Johnsville, NY
152,280 Ryan Van Manen, Grand Rapids, MI
30,380 Steve Curnow, Woodbury, MN

SPACE INVADERS *(Spectral Associates)*
1,496,030 ★Jimmy Chan, Kincardine, Ontario
62,300 ●Peter Niessen, Carlisle, MA
54,980 Susan Brink, Portage, MI
29,560 Greg Brink, Portage, MI
11,750 Bob Hosken, Mentor, OH

SPACE RACE *(Spectral Associates)*
83,750 ★John Osborne, Kincardine, Ontario
62,875 Shane Deike, Bolme, TX
59,825 Jim Baker, Florissant, MO
58,100 John Cole, King City, Ontario
55,625 Steve Skrzyniarz, Tacoma, WA

SPACE RAIDERS *(Mark Data)*
112,100 ★Jimmy Chan, Kincardine, Ontario

SPACE SENTRY *(Spectral Associates)*
41,129 ★Steve Skrzyniarz, Tacoma, WA
2,625 Cameron Amick, Reisterstown, MD

SPACE SHUTTLE *(Tom Mix)*
595 ★Steve Schweitzer, Sewell, NJ
575 Fred Weissman, Brookline, MA
571 Ted McDonald, Summerville, SC
566 Tim Smith, San Rafael, CA
562 Kenton Fifield, Ft. Frances, Ontario

SPACE WAR *(Spectral Associates)*
400,190 ★Mark Felps, Bedford, TX
116,000 Peter Niessen, Carlisle, MA
52,380 Jim Baker, Florissant, MO
16,500 Chris Leek, Martintown, Ontario
11,540 David Iverson, Dorval, Quebec

STARBASE ATTACK *(IMB)*
328,000 ★Paul Morris, Richland, VA
62,551 ●Chris Coyle, Selden, NY
60,602 Beth Walker, Gloucester Pt., VA
35,659 Dennis Teague, Noblesville, IN
24,668 J.W. Panks, Indianapolis, IN
24,553 Steve Walters, W. Lafayette, IN

STARBLASTER *(Micro Works)*
408,245 ★Mark Dowling, San Bruno, CA
325,790 Mike Anheluk, Fall Creek, OR
126,135 Mike Hall, Hartland, WI
80,001 Alan Lewis, Ridgefield, CT
35,400 Peter Niessen, Carlisle, MA

STARFIRE *(Intellectronic)*
3,444,500 ★John DeMuth, Prairie de Chien, WI
2,102,450 Dean Bouchard, Kingston, N.S.
1,320,150 Joy Bailey, Lexington, NC
1,000,000 D.D. Jackson, Ontario, Canada
1,000,000 Shaw Jackson, Ontario, Canada

STARSHIP CHAMELEON *(Computerware)*
861,200 ★Greg Lesher, North Chili, NY
82,200 Chris Sweet, Harvard, MA
79,250 Vince Lok, Mississauga, Ontario
75,500 Ron Pinson, Guelph, Ontario
72,600 Jim Baker, Florissant, MO

STORM *(Computerware)*
840,010 ★Roland Hendel, Mississauga, Ontario
723,335 Chris Sweet, Harvard, MA
472,320 John Jaworski, Nashua, NY
360,000 Cameron Amick, Reisterstown, MD
312,000 Dave Lubnow, Sussex, NJ
109,350 Kenny Miller, Yardley, PA

STORM ARROWS *(Spectral Associates)*
94,000 ★Steven Ohsie, Houston, TX
22,200 Cameron Amick, Reisterstown, MD

TEEEOFFF *(Prickly-Pear Software)*
32 ★Mark Wright, Tipton, MI

TRAPFALL *(Tom Mix)*
113,408 ★Rich Trawick, N. Adams, MI
98,588 Dan Burch, Louisville, KY
90,146 ●Neil Berkman, DeWitt, NY
89,162 Rich Van Manen, Grand Rapids, MI
85,978 Ryan Van Manen, Grand Rapids, MI
85,758 Brian Chaples, Springfield, VA
85,022 Dennis Burch, Louisville, KY

VENTURER *(Aardvark)*
2,657,350 ★Brian Panepinto, Spencerport, NY
2,152,150 ●Greg Scott, Orlando, FL
1,769,400 Todd Hauschildt, Red Wing, MN
1,526,200 Peter Niessen, Carlisle, MA
1,126,550 Hwan Joo, Weston, Ontario

WHIRLYBIRD RUN *(Spectral A sociates)*
98,400 ★Dave Lubnow, Sussex, NJ
48,000 Todd Brannam, Charleston Hts., SC
44,900 John McClennan, Holliston, MA
37,900 Mark Nelson, Alexandria, VA
30,130 Ron Rhead, Ontario, Canada

WILDCATTING *(Radio Shack)*
30,555 ★Gary Jones, Dale, TX
28,648 Lori Kozza, Allentown, PA
28,625 Kevin Marsh, Bokeelia, FL
27,455 Annie Rosenberg, Prestonsburg, KY
26,726 Stephen Jinkich, Pittsburgh, PA

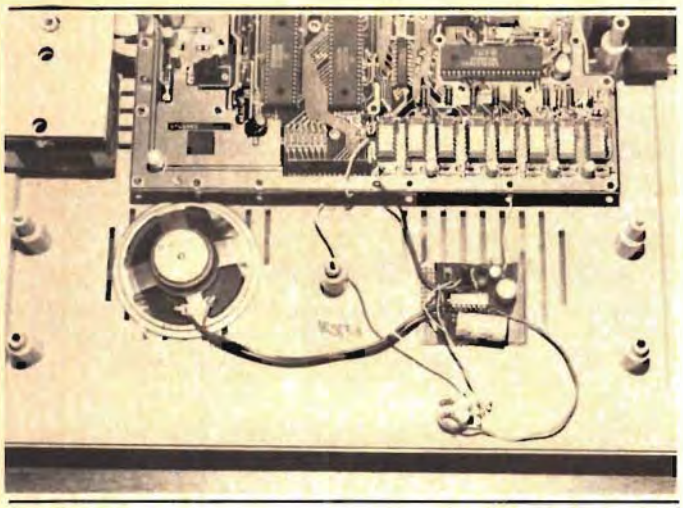
ZAKSUND *(Elite Sof ware)*
1,128,050 ★Richard Minton, West Frankfort, IL
1,008,100 Andy Mickelson, Granville, OH
950,500 ●Michael Rothman, Solon, OH
876,200 John Osborne, Kincardine, Ontario
617,500 Ed Bottini, St. Louis, MO
193,450 Andy Tait, Lexington, MA
133,600 Ron Gherman, Marion, OH
112,050 Luc Archambault, Quebec, Canada
45,900 Frankie Jimenez, Mesa, AZ
22,000 Steven Jimenez, Mesa, AA

ZAPEM *(Chromasette)*
12,950 ★Michael McCafferty, Idaho Falls, ID

ZAXXON *(Datasoft)*
2,000,000 ★Roland Hendel, Mississauga, Ontario
1,510,000 James Quadarella, Brooklyn, NY
401,900 Mike Hughey, King George, VA
370,400 Chris Coyle, Selden, NY
235,200 Rich McGervey, Morgantown, WV
134,600 Rich Trawick, N. Adams, MI
127,300 David Karam, Austin, TX
100,000 Rich Van Manen, Grand Rapids, MI
96,600 Ryan Van Manen, Grand Rapids, MI
81,300 Steven Friedman, Kings Park, NY
60,500 Damon Frazier, Bringhurst, IN

Build A Speaker/Amplifier For Your Computer

By Tony DiStefano
Rainbow Contributing Editor



January 9, 1981; that was a great day. I bought my first Color Computer. Today, two and a half years later I bought a video monitor. It is a standard composite-video monitor. It is a 9" green phosphorous screen Electrohome. I know what you are thinking, "Oh no, not another video monitor adapter!" Well, I'm not about to bore you with another version of this adapter. I used one of them myself rather than design my own. When I connected my monitor, I was delighted with the clear, crisp quality of the picture. I found that it had one thing missing—a speaker. I could not make any sounds with this monitor because it did not have a built-in speaker. At first, I would keep my color TV set next to it with the volume up. That was quite an inconvenience. Well, you guessed it, this month's project is a low cost, built-in speaker and amplifier for the Color Computer. The whole thing fits under the keyboard. It even has a volume control with an on/off switch.

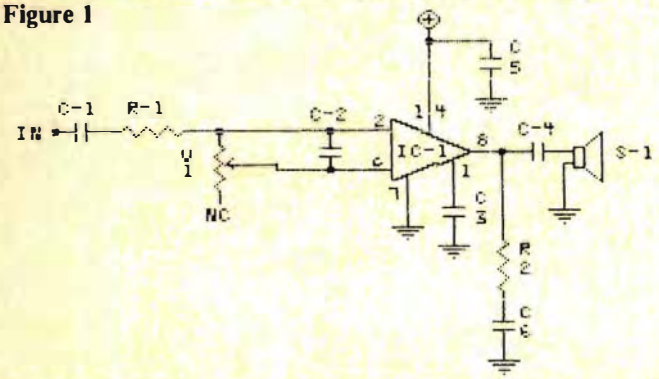
The heart of the amplifier is the power IC # LM80. This is a complete preamp and amp in one. It requires a minimum of support parts and is quite rugged. It also has a high input impedence, about 150k ohms. A high impedence is needed in our case because the sound circuit on the Color Computer is high impedence. If you look in the '83 Radio Shack catalog, you will see that they say the power supply has a maximum of 10 volts. That is not true, it must be a mistake in printing. It can, in fact, take up to 22 volts for B+. We will be using 12 volts.

To construct this project you will need the standard project tools—things like screwdrivers and pliers and cutters and soldering iron and solder and a drill to mount the volume control. You will also need everything on the list of parts. See Figure 2. All of these parts are quite common and need not be bought at Radio Shack. As a matter of fact, I had all of the parts in my parts bin. I have included the Radio Shack numbers, where possible, just as a matter of convenience. Mount all of the components except the speaker and volume control on the Proto board. Following the schematic in Figure 1, solder all the components together. All the ground points indicated on the schematic should be soldered together at one point. This is to prevent what is known as ground loops. A ground loop is when an electrical signal has two or more paths to get to the same point. This path or loop can act like an antenna, in which it

(Tony DiStefano is well known as an early specialist in Color Computer hardware projects. He is one of the acknowledged experts on the "insides" of CoCo.)

can radiate RF noise or act like an RC circuit and cause feedback. Though it is not indicated in the schematic, pins 3, 4, 5, 10, 11, 12 are also grounded. This acts like a heatsink for the IC, and should be enough for most applications, but if you think that you'll be using this amplifier very loud, it

Figure 1



would be wise to add a small heatsink to the IC. Also, make sure that the ground wire that goes from this board to the main board is at least 22 gauge. Make this wire about 4" long. Now, the B+ line (12 volts) should also be 22 gauge. This wire will go to one side of the switch on the pot. Make this wire 5" long. The other side of the switch will go to the 12 volt supply. You might think this to be heavy wire, but this chip can deliver up to 8 watts. (That is a lot of power.) The switch-to-power wire should be about 10" long. That will go to the B+ on the main board. The connections for the speaker should be 24 gauge. Make these wires about 5" long. That should be long enough to reach the holes on the other side of the computer just underneath the keyboard. Solder the other ends of the two wires to the speaker. There are two more wires from the board, and they go to the volume control—one wire to the center and the other to the left side. Make them about 5" long, too. That will be long enough to reach anywhere in the front of the computer.

Open the computer in the usual way and remove the keyboard. Place the components in accordance with Photo #1. You may want to tape them down temporarily so that they don't move around too much. Drill a hole in the computer to mount the volume control. Personal taste will judge exactly where to drill it. The hole should be $^{5}/_{16}$". Mount the volume control in the hole. Be careful not to break the attached wires. Make sure that the position of the volume control will not get in the way of the keyboard. The next step is to connect the B+ (12 volts) and ground. If you have the "F" (or 285) board, find the power by looking at the photo. It is the jumper for 16/64K memory. Use the one marked 16K. For the ground connection, scratch off a bit of the green coating on the PCB just to the right of the keyboard connector, under C59, and solder to that. If you have another version, use Test point #9 for the 12 volts and Test point #4 for ground. The last connection to make is the input. That connection goes all the way to the top. It connects to pin #3 on the RF adapter. This is all that has to be done; but before you close the computer, check your work.

Replace the keyboard and turn the computer on. To test your amplifier, any sound command will work. This one line program works fine:

```
10 SOUND RND(255),1 : GOTO 10
```

Turn the volume control on. You should hear a click. Turning the volume control up should result in some random sounds coming out of the speaker. 100 IF SOUND=NONE THEN TROUBLESHOOT ELSE CONTINUE. Only kidding folks, but that is the next step. If you don't get sound, check your wiring and check for cold solder joints. Make sure that the chip is plugged in the right way. If the sound is loud at first and drops as you turn the volume control up, you have the outside wire on the pot on the wrong side. Unsolder, and reverse it. Other than that, you should have no problems.

Now, you could leave the speaker and amplifier just taped down, or you could use some rubber cement or screws. Don't use a permanent glue though, it could make a mess if ever you have to remove this thing from the computer or change the speaker. When all checks out, replace the cover and enjoy your new speaker and amplifier.

Figure 2

Parts List

#	Part	Description	RS #
R-1	Resistor	150K ohms	271-047
R-2	Resistor	2.7 ohms	n/a
C-1	Capacitor	10 uf @ 16v	272-1423
C-2	Capacitor	.022 uf @ 16v	272-1066
C-3	Capacitor	10 uf @ 16v	272-1423
C-4	Capacitor	470 uf @ 16v	272-957
C-5	Capacitor	220 uf @ 16v	272-1006
C-6	Capacitor	.1 uf @ 16v	272-1069
S-1	Speaker	2 to 5 inch	40-248
IC-1	Amplifier	LM 380	276-076
PC	Proto-Board	.1 inch spacing	276-1392
V-1	Potentiometer	100K ohms	271-216

RAINBOW REVIEWS

REVIEWS

SEA DRAGON

Editor:

Adventure International would like to comment on the review of Sea Dragon in the September issue by Neil Edward Parks. Normally, we would be reluctant to do so, but we felt that the tone of the article and the conclusions drawn by the reviewer do not remotely reflect the high quality of Sea Dragon.

Clearly, two-thirds of the review deals with items that have nothing to do with game play, and by the author's own admission, he states that "Even in the Practice Mode, I've never made the first plateau." This, I would suspect, would be the reason so much time was spent with what (game play options) Mr. Parks states are "major faults." Sorry . . . but the game does not hang up in the middle of play, nor does it do less than the advertising promises. If this was the case, we would agree that Sea Dragon has major faults. Mr. Parks rightfully points out that there was a problem with how the options work . . . but, we consider this a minor fault. In fact, we have yet to receive a single complaint and if Mr. Parks would have given us a call, he would have found that the problems had been noted by Adventure International and were fixed.

Mr. Parks also states that he is not "enamoured of the packaging" and finds it bulky and "awkward." He is certainly entitled to his opinion, but his opinion does not reflect the fact that Sea Dragon is sold in the best packaging available in the industry and was designed for ease of use and durability. Adventure International buys its hardboxes in quantities of 50,000 to 100,000 units. Therefore, the cost of our software is not a reflection at all of the high quality packaging we use. Instead, Mr. Parks should consider the high cost of doing business and ask what part the cost of advertising, printing, artwork, salaries, color separations, royalties and overhead play in determining the final cost of software. I assure you it is considerable.

Adventure International acknowledges the minor problems noted by Mr. Parks, but we feel he may not be suited to stand in judgment of a product that he, by his own admission, cannot play. We, at Adventure International, wonder how one can reasonably criticize a piece of software when less than one-tenth of Sea Dragon has been played.

Mark Sprague
Product Development Manager
Adventure International, Inc.

VOCABULARY BUILDERS

Editor:

Our 32K program, *Vocabulary Builders,* was reviewed in the September issue of *the Rainbow*. We appreciated the kind words that were written, but feel that several of the criticisms were humorous at best. Great care, preparation and much field-testing go into each Computer Island program.

The reviewer's main criticism was our method of randomly selecting the questions used. I have described, in several of my articles for this magazine's educational column, various ways of randomly selecting questions in a program. Our *Vocabulary Builders* program selects 50 out of 200 questions randomly for each set. On the next series of questions, the program selects again 50 of the next 200 questions. This guarantees that some questions will be repeated and some will be new. The reviewer suggests that some of the 200 questions may not be used after many, many rounds. Exactly our purpose!

We extensively field-test all of our programs and have found that children easily memorize answers in short reading and language arts programs. After several rounds, it is often unclear whether the child has mastered the work or merely memorized the location of specific answers. Our method of presentation insures that new words will keep cropping up set after set of questions to prevent the memorization factor.

The reviewer also felt that we should have used *INKEY* instead of *INPUT* as the method for having answers entered. We though of using that method but found it to be unwise in this case. When field-testing this program, we noticed that children often hit the wrong answer key by mistake. *INPUT* gives the child a second chance by allowing him to use the back arrow and change his answer before hitting [ENTER]. We are looking for accuracy, not speed, in this type of program.

Finally, one of the main strengths of our *Vocabulary Builders* program is the amount of questions it contains. Children use these programs to prepare for various types of system-wide school tests. The more questions and examples covered, the better prepared they will become: We met that need by preparing 32K programs packed with *DATA* and examples. Our reviewer's suggestion that our program could be reduced to 4K finally reduced all of our credibility in his credentials.

We truly wish that the reviewer had spent more time discussing the degree to which our program actually teaches vocabulary. An educational review is not the proper place to state one's philosophy of education or debate programming techniques. It should be the place to report whether the program works well on the children for whom it was intended. In a two-page review, the reviewer made no mention at all of his using our programs with any children.

Steve Blyn
Computer Island

TALKING SPELLER

Editor:

I am writing in regards to a review of *Talking Speller* published in September 1983 *Rainbow*, page 199. We are always happy to see one of our products reviewed, but in this case the review requires comment. I am not critical of that portion of the review directed to the actual performance of the *Talking Speller*, as I feel it was very adequate. I do object strongly to the inclusion of editorial comments into a review, as this destroys any pretense of objectivity on the part of the reviewer. In this case, Professor Tchudi has his own concepts of the proper use of computers in education. That is his right. However, since the same issue of *Rainbow* carried an article by the good professor ("The Write Idea," page 139), I believe your readers would have been better served had he confined his philosophy lecture to his own article. Since he did not, I feel compelled to reply.

Granted, use of a word processing system and a spelling checker may well be the best method to really master spelling—for those students capable of utilizing such educational aids. However, in this part of North Carolina, we have very few first-, second- or even third-graders who know how to type. Many of them do not even know how to operate a computer, much less use word processing. Perhaps elementary students in Michigan are so advanced that Professor Tchudi's theories apply. In our case, they are not necessarily valid. Many of our elementary teachers do not feel that spelling drills, of whatever form, are "trivial" or "pedagogically obsolete." Instead, they are convinced that a student must learn proper spelling of a few words, by whatever method, before the students can be launched on their writing career.

Professor Tchudi missed the point completely. *Talking Speller* is aimed at the preschool and lower elementary levels. At this level, regardless of how pedagogically advanced the educational system, you cannot teach advanced concepts when the students do not have the basic motor skills.

Happy Bits and Bytes.

H. A. Manning
Superior Graphic Software Products

Colorful Budgeting With *Color Finance*

By Frank J. Esser

Finance programs on today's market run from the simple programs to balance your checkbook and provide information on where your money goes to those that provide a more complete financial picture. *MSI Color Finance*, from Micro Services, Inc., and distributed by Delker Electronics, is a program that gives you the ability to better manage your personal finances. Properly set up and maintained, *MSI Color Finance* can give you an accurate picture of your finances at any time during the year with surprisingly little effort. *Color Finance* is set up as a double entry bookkeeping system, tailored to a personal finance application. For those not familiar with double entry bookkeeping systems, do not despair, *Color Finance* has excellent menus and documentation to take you through its setup and use.

Color Finance comes on a single 5¼-inch diskette and requires a Color Computer with 32K of memory, disk drive and a line printer. Optionally, a cassette recorder can be used to provide backup of the data files. The disk is not copy protected. However, *Color Finance* does have a safeguard which will discourage unauthorized access to your financial records. A plug is supplied which must be inserted in the left joystick port before *Color Finance* will run. A neat way to discourage tampering with your records—like the lock on your personal records file.

The documentation is of excellent quality and comes bound with a plastic spiral binding. The manual is well laid out and very easy to follow. It is divided into four parts: Introduction, Getting Started, Commands and a Sample Session.

The introduction gives a brief overview of the program and its capabilities. Part I—Getting Started is just that. This chapter takes you through the steps necessary to get *Color Finance* up and running. The account codes are explained along with their classification. There are three account classifications and a range of code numbers associated with each. They are:

01-21 Asset Accounts
22-42 Liability Accounts
43-96 Expense Accounts
97 Income Account

Several examples of each account type are given. For instance, checking and savings accounts are assets, while charge accounts and bank charge cards would be liabilities. Items such as food, rent, gasoline, clothing and heat would be expense accounts. Also explained are the 75 descriptive categories available. These categories are used to help explain each entry made into the system. I found that, in my personal application, I used only half the available entries. There is also a table explaining how each type of entry affects each class of account.

Part II—Commands is the heart of the manual. It is set up such that it follows the menu presented at startup. There are 10 selections presented on the main menu. They are:

1) Daily Posting Module. Provides the ability to add entries, print account statements and exit this section. This module handles the entries for a given period, in this case a month.
2) Monthly Update Module. Updates your financial records for a given month and prepares the balances for the upcoming month. Will also print description summary, general journal entries, trial balance, account statements for any given month.
3) Yearly Module. Provides the ability to obtain on a year-to-date basis the following reports: budget, description and the accounting statement.
4) Account Maintenance. Provides the ability to enter the account titles for each of the account classifications along with the beginning balance and the month that balance is valid for.
5) Description Maintenance. Provides for the entry of description codes to be used during actual data entry.
6) Change Today's Date. Allows for the entry of the date which will appear on the heading of each report generated during this session.
7) Utility Module. Provides for the ability to backup or restore your records for cassette tape.
8) Codes Listing. Will produce the account names and code numbers by account classification. Will also produce a listing of all the description codes and code numbers.
9) Initialization. Provides for the intialization of the data files for startup or at the beginning of each new year.
10) Exit System. Leaves the program and returns to the BASIC system.

Part III—Sample Session provides a brief session using the data files already existing on the supplied disk. This session walks the new user through six entries for a given month, explaining each entry in detail. You are directed to

this section in the first chapter *before you initialize* the disk. The initialization process will wipe out the data in the files, thus making the sample session impossible.

Color Finance is a well-designed and smooth running program. The menus are excellent with clear crisp lettering done in the graphics mode. All the menus used by *Color Finance* are done on the graphics screen, thus they are not limited to the standard 32 x 16. Instead, the screens are 42 x 32. The options are always explained and a prompt line appears at the bottom of each menu handling data entry or correction. If an error is made it is easily corrected.

Color Finance is a complex program and, being such, requires more than the average effort in becoming familiar with it. However, if you are willing to put in the time and effort I am sure you will find the investment well worth it.

How does *Color Finance* provide these financial services and how does it work? *Color Finance* is a double-entry bookkeeping system tailored to a personal finance application. A double-entry bookkeeping system is one in which a corresponding entry is required to offset the original entry. Thus, at any one time, the summation of all the debits will equal a summation of all the credits, indicating that the books are in balance. Setting up the accounts for *Color Finance* is simple and really depends on how complicated you want your application to be. If you simply want to track your monthly expenses and not cover your net worth or charge accounts, then all that is required is a list of your monthly expense items and descriptions of each. However, if you want *Color Finance* to track your entire financial picture, then much more information will be needed. You will need to know what assets you own and to whom you owe money. You will also need the above mentioned expense lists to complete the list. Items which would fall into the asset category would be, cash on hand, either in savings or checking accounts, cash value of life insurance policies, saving and investment plans, and IRAs. Examples of liabilities would be a bank loan for an automobile, charge accounts, bank credit cards and any other open loans. Expenses would be items like the mortgage, clothes, food, auto, recreational activities, medicine, doctor/dentist, insurance and utilities.

I think by now you are getting the picture. You are now ready to create your accounts and descriptions. Data is entered from your checkbook stubs on a month-by-month basis. You can enter a month's worth at a time or as many months at one sitting as desired.

I think I write more than the average number of checks a month, between 50 and 60. It took a little more than 45 minutes for me to get them entered for a one-month period. The reports ran a little over an hour and 15 minutes. Not bad considering it needs to be done only once a month.

Some of the hardest items to budget for and track are charge cards. Most systems do not make provisions to handle items purchased on time easily, if at all. It is here that *Color Finance* really shines. At the time the purchase is made, the expense account describing the item is debited and the appropriate liability account covering the charge card is credited. When the first of the month comes around and it is time to make a payment, then the checking account is credited and the liability account is debited. This has the net effect of showing the item purchased and in its appropriate category at the time of purchase for its full amount. It also reduces your net worth since you do not yet own the entire item. As you make the payments, your net worth will increase because your liability is decreasing. I suppose all of this sounds complicated and perhaps to someone without any bookkeeping or accounting experience, it is. The real secret of *Color Finance* is becoming familiar with it and how it works and then tailoring the setup to your needs.

Color Finance is everything it is advertised to be, and then some. I found it easy to use, with good documentation, excellent menus and very good error trapping. *Color Finance* uses a double entry bookkeeping system to maintain its records and is set up in such a manner as to prevent off balance situations from occurring.

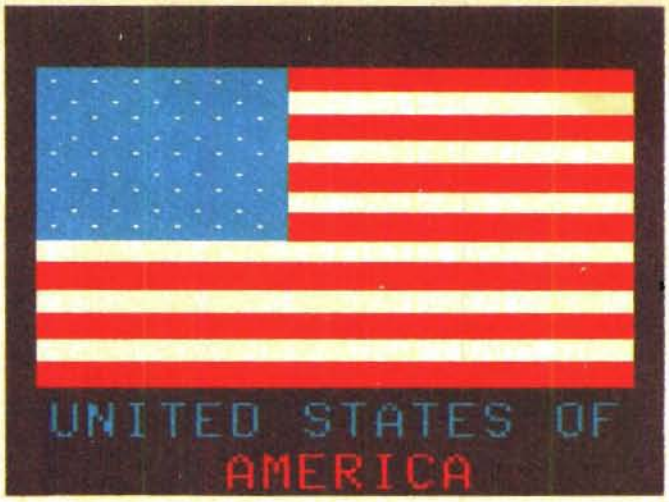
The program provides for full maintenance functions such as add, change and delete records. It has several print functions such as account statements, budgets, trial balance, and general journal entries. It also has a backup-restore function to cassette tape—something I wish more developers of business software would provide. The ability to store data on tape is a definite plus since the tape storage is much more rugged and versatile than floppy backup, not counting the economics.

Color Finance provides for 22 asset and liability accounts, 56 expense accounts and 75 separate descriptions. The only problem I had with *Color Finance* was the description fields. They are 10 characters wide and I found this to be a little too restrictive. A field width of 20 characters would have made for much more meaningful entries. However, this is a minor item on what I consider an excellent program.

If you are looking for a system to record and update your financial transactions, the CoCo and *Color Finance* are a winning team.

(Delker Electronics, P.O. Box 897, Dept. D, Smyrna, TN 37167, $59.95 on disk)

Color Ink Jet Printer Is Sensational CoCo Peripheral



There are some things that are just so sensational that it is difficult to keep from being excited about them. Radio Shack's newest printer — the CGP-220 Color Ink Jet printer — is one of them.

I must confess that it isn't very often that I get the opportunity to do reviews anymore. Sure, I spend a lot of time looking at both software and hardware products as they come in, and more time looking at the programs which are submitted to be included in *the Rainbow*. But to be able to take the time out to put a piece of hardware through the paces? Well, I *do* wish I had a little more time for that.

But, from the time I bought my first Quick Printer II for my first CoCo, printers have been my "thing." And then along comes something that is absolutely revolutionary for the Color Computer. So, I was hooked. Am hooked. And, if you're into color graphics, then I think you'll be hooked, too.

The Color Ink Jet Printer prints by using two reservoirs of colors — one of them black and the other red, yellow and blue. Now, as anyone will tell you, those three colors, when combined, will give you three more colors — orange, purple and green. Add the black and non-printing, or white, and you suddenly have a printer which prints with eight colors!

Yes, the Color Ink Jet Printer *automatically* combines red and yellow to create orange. All you have to do is give it a command. Needless to say, this can really do some interesting things for your listings and printouts — you can color-code them. And in two sizes, too.

This has some real possibilities, all in itself. I can't wait for someone to come up with a program that not only "pretty prints" listings, but changes the colors, too.

But, this isn't where the Color Graphics Printer really shines. It does that with its high resolution color graphics. And, if you think *PMODE4* has high res, wait until you see the Color Ink Jet! It generates 640 dots per line — a higher resolution than CoCo itself.

We ran an article several months ago by Dr. Lane Lester which showed how you could make color pictures on your printer using colored ribbons. The Color Ink Jet lets you print the same quality pictures — or better — in one fell swoop! And what pictures! See the example here, and elsewhere in this month's *Rainbow*.

I was telling you about my first printer, the Quick Printer II. Now, let me tell you about my second, the Line Printer VII. Among everything else my kids remember about my computing when *the Rainbow* was still a "kitchen table" operation was the noise the LP VII made while it was printing out copy and listings. It is still worth a good laugh to hear the girls' imitation of the noise — *BURRRRUP, CHUNNNG!*

Color Ink Jet won't even wake the mice. Because the technology is that it actually sprays individual drops of ink on the paper, there is, for all practical purposes, no sound at all.

It features bi-directional line scanning, a 5x7 dot matrix for the letters and true descenders. It handles both sheet fed and roll paper in full 8½x11 size. Color Ink Jet also has both parallel and serial interfaces and, in serial mode, will accept Baud rates of both 600 and 2400. A simple switch on the back of the printer and a *POKE* to CoCo allow the faster speed.

Yes, there is a drawback — speed. In text mode, Color Ink Jet prints at only 37 characters per second. In graphics mode it will handle 2630 dots per second. The reason for this seeming difference in speed is that it forms its letters graphically — by printing across a line three times to fully form a letter. So, although the printhead is moving pretty quickly, it takes more than one pass to actually form a letter.

Those of us used to just turning on a printer and telling it to print will have to get used to one more thing. The Color Ink Jet Printer has a "new" lever which you use to cap the ink supplies. This keeps the inks from spilling when the printer is being moved and, more important, keeps them from becoming dried out. If you've ever put a brand new ribbon in your conventional printer and then gone off on vacation two days later only to find on your return that the ribbon has dried out, you'll really appreciate this feature. And, you can use the same lever to "pump" the ink supply to darken it up (a little) if you wish.

Setup was extremely easy and the instructions were clear, well detailed and easy to figure out. We were printing out some color graphics about 30 minutes after we opened the box.

And now for the kicker. Radio Shack also has a new screen dump program in machine language written for the Color Ink Jet. With it, you can select your background color and you can print from all PM ODEs. This is extremely easy to use and provided some excellent color printouts. There are also "standard" graphics characters for the "other" Radio Shack computers.

All in all, we think this is an excellent printer with some very exceptional capabilities. The colors and graphics are outstanding, the alphanumerics are well-formed and the lack of noise is a real bonus. Combined with ease of use and some truly exciting possibilities the Color Ink Jet Printer has to offer, we believe this one to be a real winner at an attractive — for an ink jet printer — price of $699.

(Radio Shack, available nationwide, $699)

— Lonnie Falk

It's A Space Adventure: But, Inner Or Outer?

Among the better offerings of computer games for CoCo grenadiers introduced in recent months is *AREX*, a program adapted for our computer by highly-talented Roger Schrag and marketed by Adventure International of Longwood, Florida.

Although billed as a space game, *AREX* could just as easily have been named something like "Super Squares" or "Blockbuster" because, in the course of the game, you're involved with trying to fill 90 percent of the screen with squares. The only resemblance to a space game is a tiny but deadly missile whose unpredictability will have you banging on your keyboard in frustration.

Accompanied by a nice assortment of sound effects, *AREX* greets you with a nicely designed title page, asking "1 or 2 Players?" You have an option of using joysticks or the arrow keys on your computer.

Your mission as the *AREX* Commander is to capture 90 percent of the enemy's territory by skillfully maneuvering the vehicle horizontally or vertically. As the ship moves, it "neutralizes" enemy ground, leaving a trail of squares. When you have conquered the territory on each level, there's a temporary pause before you advance to the next level.

On the various levels, you encounter a variety of enemy ships: There's the Snarfer, which moves at right angles only, but with an ability to accelerate rapidly. The Diagon, which I had the most problems with, moves in highly unpredictable patterns and is without a doubt the most dangerous of the enemy defenses. The Diagon appears to have radar and is a very capable defender, so watch him! The Rippler is more an effect than an intelligent danger. It appears only when your ship is stationary for more than five seconds, tracing the path of your vessel until it wipes you out. You can expect to encounter the Rippler when you have blocked yourself in.

Points are earned in three different ways: Each square you occupy in enemy territory is good for four points; the figure increases by 10 with every four successful units. Ramming a defenseless Snarf is good for up to 900 points. Knocking off the Snarfs gives you a great sense of accomplishment because there are some nice sound effects. Walling an enemy ship, which is accomplished by trapping it with the trail of your vessel, is worth 900 bonus points. Thus far, I have found walling all but impossible because the enemy moves so quickly that you can easily find yourself being put out of commission. The task is made even more difficult by the fact that the area of confinement must not be more than the size of one area unit.

Enemy ships will sometimes collide during the course of the game, causing them to exchange battle characteristics, or they may be destroyed. An innocent Snarf, for example, will suddenly be transformed into a deadly Diagon, or vice versa.

You can earn an extra ship for every 10,000 points you rack up.

All but the first level, which is completely clear except for enemy craft, have different physical barriers that you have to negotiate while attempting to earn points and stay alive. The barriers and the unpredictability of the enemy make *AREX* a game of continuing challenge, and, therefore, a game you will not tire of after a few tries or a couple of weeks.

Another nice feature is the ability to record high scores. Initially *AREX*, with 3,000 points, holds the top 10 positions. The low point total built into the score columns is a nuisance, too, because it is so easy to get 3,000 points that you are constantly typing in your name. I resorted to just hitting the space bar to keep the game moving unless I was able to get a real high score. After you turn CoCo off, of course, the scores are erased. The documentation, which is quite good, indicates that scores can be saved on the TRS-80 Model I and III versions of the game. Oh, well.

Another feature that the other models enjoy is voice-announced information. A call to Adventure International led to the conclusion that voice was unavailable for CoCo because it could not fit in 16K models, and the widest market for such games is for such owners. (Someday there'll be some super fantastic programs for us folks who have upgraded to 64K.)

A must for any review of *AREX* has to include a word about the high quality packaging of the program. A full-color package with super graphics simply jumps out at you and demands that you have a copy. Someday, too, there will be retail stores everywhere with third-party software (hopefully including Radio Shack outlets) so that such packaging can realize its potential. Adventure International is to be commended for its professional approach.

(Adventure International, P.O. Box 3435 , Longwood, FL 32750, 16K ECB, $34.95 on tape)

—Charles Springer

COLOR TERM +PLUS+ Has Many Plusses

By Bruce Rothermel

Let's take a few seconds to consider the wonderfulness of the free enterprise system.

We are all free to enter the consumer market and introduce and sell "The Better Mouse Trap." Whether or not the people beat a path to your door depends on how good the product is, the support given to the product, and, of course, the price.

An excellent example is the Terminal Program Market for the 80C.

For those of you new to the principles of Data Communications via the 80C: It is possible for your computer to "talk" to distant computers using telephone lines. These telephone lines were not designed to handle data from a computer which consists of a series of "bits" of information, where each bit has only two values, on or off—a digital signal.

The telephone line was designed for the many different tones that make up the human voice—an analog signal.

For transmission through the telephone line, the digital signal is changed to an analog signal by means of an electronic converter called a "modem." The modem converts (MODulates) the digital signal into an analog signal compatible with the telephone line. At the other end, another modem converts (DEModulates) the analog signal back to its original digital form.

Modems are available which operate at many different data rates measured in bits per second (BPS). The higher the rate, the faster the information is transferred.

Terminal Programs allow the 80C to handle the connection to the other computer, and direct the flow of data to and from the two machines through the modems.

Most reasonably priced modems operate at 300 bits per second — 300 Baud. Higher priced modems operate at 1200 Baud and up.

In the beginning there was *Video Text 1.0* from our friends at The Shack. Using this ROM pack and a modem, the CoCo could easily link up with a bulletin board or data bank service like CompuServe.

However, *Video Text 1.0* did not allow the user to easily print the information received from the service or bulletin board and programs could not be transferred between machines. When *Video Text 1.0* was written, the 32K 80C did not exist and CoCos were treated as if they were 4K machines with virtually no usable buffer memory. While the 32K problem was solved with the introduction of *Video Text 1.2*, the lack of printing and transferring programs was not addressed. Even the revised issue was a dumb terminal.

For those of you still stuck with version 1.0, and having 32K computers, the 2.0 version is available for an upgrade charge of $15.95 (order stock #700-3301).

To our rescue came the Independent Programmers—and now we have a selection of "intelligent" terminal programs to choose from. This strong secondary market has even created competition among these programmers and as a result we are getting better programs, with more features, for less money.

Ain't free enterprise wonderful!

The subject of this review, *COLOR TERM +PLUS+*, from Double Density Software, is one of those Intelligent terminal programs.

COLOR TERM +PLUS+ has been upgraded to include additional features and capabilities. You can now:

Select communications Baud rate from 110—19200
Select printer Baud rate from 600—9600
Select Half or Full Duplex
Select Odd, Even, or Full Parity
Select 7- or 8-bit words
Send Control Characters
Turn off lower case letters
Word Wrap—on or off
Select normal or reverse video
Scroll protect up to nine lines
Automatically capture incoming files
Send one line at a time from your buffer
List directory and granules (disk version)
Show buffer size remaining
Up and Down load programs
Edit buffer
Scroll buffer
Load and unload the buffer when off line
Save and load machine language or BASIC programs or files
Pre-enter data before going on line

While this list at first boggles the mind, with a little practice, using the program is easy. The above options allow you to configure this communications program to meet your needs.

The activities involved in a typical communications session using *COLOR TERM +PLUS+* would be:

1) Load ROM Pack, tape or disk version.
2) Set Parameters—These are preset to 300 Baud, Even

Parity, Half Duplex, 7-bit words, Word Wrap Off, Lower Case Letters Off, No Auto Buffer.

If the computer you are going to communicate with is set to the same parameters, go on. If not, they can be easily changed by answering the prompt screens. With CompuServe for example, just change Half Duplex to Full Duplex and Log-On.

3) Communicate Mode—Log On can be either automatic by pre-loading your ID number and password into your buffer for transmission, or by typing the information directly on line.
4) Storing Information in Your Buffer—The Buffer is the storage area set aside in memory in order to store information the user may wish to save or print later. The buffer can be turned on or off to save only the information you will want to retrieve or print later. For instance, when accessing "World Book Encyclopedia" on CompuServe, you would close the buffer during log-on and then open it to keep the subject information in the buffer for later review or printing.
5) Transmitting and Receiving Programs—In addition to capturing data, *COLOR TERM +PLUS+* allows the user to transmit and receive programs from one computer to another by preloading a program into the buffer and transmitting it over telephones, or receiving a program into its buffer. BASIC, ASCII and machine language programs, data and files can be transmitted in this manner.
6) Editing the Buffer—After this info is loaded or received into the buffer, if it is a BASIC program or file, it can be edited. The editor is not sophisticated, but very usable.

You can:
Insert a single space
Delete a single space
Delete a word or part of one
Delete an entire line
Change any character by typing over it
Scroll up, down, left or right

7) Saving the Buffer—After the buffer has been loaded, you can go "off-line" to save long distance fees and service charges; then you have the option of dumping the buffer to a printer, saving it on tape or disk, displaying it on the screen for review or coding it for security purposes prior to saving it.

Double Density Software has priced *CT+* at $29.95 for the tape version #2.2, $39.95 for the ROM pack version #2.3 and $39.95 for the disk version #3.2. These are priced somewhat lower than other competitive communications programs.

And in the spirit of free enterprise, D.D.S. is offering a "trade-in" allowance of $10 if another original and complete program is turned in at the time of order. Quite a value! For owners of previous versions, upgrades are available for $12.54 (tape) and $17.54 (disk).

In summary, *COLOR TERM +PLUS+* offers the 80C user, who is willing to invest some time in studying the manual and learning to use the program, versatility and flexibility at a very good price.

(Double Density Software, 920 Baldwin Street, Denton, TX 76201, $29.95 tape, $39.95 ROM pack, $39.95 disk)

Disasm's New Approach Has Reviewer Longing For The Old

An assembler that uses decimal input and output instead of hexidecimal? It calls for some serious consideration before answering.

DISASM (disassembler-assembler) by Dynamic Electronics Inc. is a new approach to assembling and disassembling machine language programs. It uses decimal notation for all addressing and data. This may sound appealing to the inexperienced, (not having to learn that confusing hex notation) but it has very serious drawbacks. Namely, all good books and articles written on the subject of ML (machine language) programming are written using hex notation. Look into back issues of *the Rainbow* for articles by Lewandowski and Roslund and you will see that none, repeat, none of their articles on ML use decimal notation. They are all in hex notation. Books like Leventhal's "Assembly Language Programming" or Staugaard's "6809 Microcomputer Programming and Interfacing" use hex notation, not decimal. This does not mean that decimal notation is not possible, but where would you turn to if you needed additional information on a particular problem you had with the assembly?

DISASM is written in BASIC and is about 8K bytes long, therefore requiring a 16K system. Extended BASIC is not required.

DISASM	**STANDARD**:		
41014 LDD I 65332	A036	LDD	#FF34
41017 STA X DIR R+5 OS= 2	A039	STA	2,X
41019 STB X DIR R+5 OS= 1	A03B	STB	1,X
41021 STB X DIR R+5 OS=3	A03D	STB	3,X
41023 LDX I 65312	A03F	LDX	#FF20
41026 CLR X DIR R+5 OS= 1	A042	CLR	1,X
41028 CLR X DIR R+5 OS= 3	A044	CLR	3,X
41030 DECA	A046	DECA	
41031 STA X DIR R+0	A047	STA	,X
41033 LDA I 248	A049	LDA	#F8

The assembler is not a two-Pass Assember. It does not support labels. All locations for branches and jumps must be known ahead of time. This is acceptable if the branch or jump is backwards because you know what address you want to go to. But if the branch is forward, you don't know where the forward address is going to be. The way I found to get around this is to put in a dummy address into the branch location and then going back and inserting the address later when it was properly determined. Thus in effect, you the user must create a second pass on the assembler to complete the branch or jump instruction.

In the instructions, the first example program caused a havoc until the error was found. The instructions assembled the sample program in memory address 13000. This overwrote the existing BASIC program and caused it to crash. Moving the program lower in memory by doing a *PCLEAR 1* didn't help because the system variables were being overwritten. I then did a *PCLEAR 4* and assembled the test program into address 3000 and the sample program ran OK.

To *EXECUTE* the sample program, the instructions said to assemble a JMP E 13000 (JMP E 3000 in the modified location) into address 274. My CoCo has Extended BASIC and location 274 is where the *TIMER* variable is stored. Therefore putting a JMP instruction there was hopeless as the *TIMER* variable is always changing. The way to execute the sample program was by doing a DEFUSR0=3000 followed by A=USR(0).

When the assembler asks for an instruction and the instruction is invalid, *DISASM* prompted with "INSTRUCTION?" again, but it did load a code into a memory location causing the ML program to crash.

An attempt is made to keep the screen clean during assembling by going back and erasing previously used prompts. But the previous prompts are not fully erased and at times the screen looks like this:

INSTRUCTION?

ENTER ADDR

This gives the impression that *DISASM* wants you to input an address when in fact it wants you to input an instruction.

An instruction like: LDA X DIR R+0 is broken down into three parts by *DISASM*. Each part is processed by the program individually and the user must wait between each part while the program is processing it. This makes entering the ML program quite time consuming and uncomfortable as you have to look at the screen between each part to see if it is completed.

Overall, I do not recommend *DISASM*. I found it difficult to make the constant conversions to decimal (although hex could be forced by using &H) and the break from the traditional assembler/disassembler format was not found to be of any advantage.

—AL Burzynski

Blackjaq—A Blackjack Trainer

By Thomas C. Roginski, Ph.D.

I have been playing blackjack for a number of years and am co-author of the only book specifically for Atlantic City blackjack *(Playing Blackjack in Atlantic City).* I also am a loyal CoCo user and am therefore, very glad to review one of the two CoCo programs which try to teach, rather than only play, blackjack (the other program was reviewed in August). Blackjack is a much-studied game because it is the only game of "chance" played in the casino where the player can get the edge on the house. Blackjack experts have made a lot of money beating the house at their own game. If the game can be beaten, then why do the casinos offer blackjack? The reason is that few people know how to win. Casino buildings were not built by winning players but by losers. Blackjack has grown to be a major casino favorite since it was found to be beatable in the early 1960s. By learning how to play correct blackjack, the player can have fun and profit from the casino's money.

Blackjaq is one of the few computer programs which attempts to teach blackjack, is the least expensive, and is the only such program which can run on a 16K CoCo. The ability to make such a complex program run on 16K is an example of efficient programming technique. The 16K limitation did prevent the use of graphics in the program.

One of the nice features is that five players can play at one time. When less than five players are playing, the computer will play the vacant seats (even all five places if desired). Up to two of the players can use the joysticks to play the game. While it took a little getting used to, it was easy to bet and play entirely with the joystick. This also felt real because all play in Atlantic City is with hand signals.

In order to win at blackjack, it is necessary to both learn correct play and to learn to count. Of these, the most important at first is to learn correct play. Because no references are given in the documentation that comes with *Blackjaq,* I will suggest *Million Dollar Blackjack* by Uston, *Professional Blackjack* by Wong and/or perhaps *Playing Blackjack in Atlantic City* by Chambliss and Roginski. One of these or one of the other modern blackjack books are needed to begin to win because the casinos have changed the game to beat professional players since many of the older books were written. One of these changes is the almost universal use of multideck games in the casinos. While I would have expected *Blackjaq* to have included the single deck game as one of the options, as the authors point out, there are few single deck games left.

In addition to knowing how to play the hands, in order to be a winning player you must know when to bet high and low. In blackjack you use systems of "counting" to determine your advantage. You bet high with advantage and lower without it. In this program two systems of counting are shown. One of these systems is called HI-OPT I and is displayed as the second number at the top of the screen (the first number is the number of cards played). In HI-OPT I the cards 3, 4, 5 and 6 count as plus 1 when they are removed from the deck and the ten cards (10, jack, queen and king) count as minus 1. The higher the count, the higher the player's advantage. The third number at the top of the screen is the HI-OPT II system. In this system the 2, 3, 6 and 7 count as plus 1, the 4 and 5 count as plus 2 and the ten cards count as minus 2. All cards not listed count as zero. As an added refinement, a separate count of aces may be kept. In order to determine your advantage, these counts, called the "running counts" must be converted to the "true counts." The true count is the running count divided by the number of decks remaining unplayed. The true count times 0.5 minus the house advantage (usually about +0.5 percent) gives the player's advantage in percent. As an example, consider an Atlantic City six deck game with five decks left in the box ("shoe") and a running count of +10. The advantage to the player is then +0.5 percent. This is from ((+10/5)*0.5)-0.5=0.5 percent.

When you first begin to count, it is useful to have the count show on the screen. After a while it would be better to have a way to turn off the count, because you must learn to count by yourself in the casino.

Blackjaq does not have this option. The easiest way I have found to add this option is as follows: *EDIT*660 by hacking off (H) everything after the *INKEY*$, add a new line as follows: 665 IFA$=""THEN 660ELSE IFA$="Y"ORA$ ="N"THEND0$=A$:GOTO660ELSE B=VAL(A$):IFB = 0THEN660ELSEBE(1,0)=B:BE(1,1)=B:RETURN.

Then, change line 560 to read as follows: 560 RC=Y *RW:PRINT@0,CL;:IFD0$=""ORD0$="Y"THEN PRINT @5,INT(Y*CQ)""INT(RC);:RETURNELSEIF D0$="N" THENRETURN. Using these changes when you are asked for a bet, if you enter "N" you will freeze the display of the count until you answer "Y" for a bet. In either case, you then bet in the normal manner after changing the display.

As with many areas which are on the borderline of science and art, there are a number of disagreements about many details. The counting systems offered above use the HI-OPT systems which allow the ace to be equal to zero. The ace has a neutral value for playing the hand after the bet because it has the function of both a high card or a low card. Before making the bet, however, the card is very important because the bonus payment for a natural (blackjack) and because of several useful options in doubling of soft hands. These are hands where the ace counts as 11.

Note that if your hand does not exceed 21 with the ace counted as 11, no card which you then take will make the hand exceed 21 or bust. This makes it possible to double certain hands against the low dealer up cards which would otherwise bust.

These facts of the HI-OPT systems make them have what is called good "playing efficiency" but lower "betting efficiency." The simplest system with the ace counting as other than zero is the HI-LOW system which has the 2, 3, 4, 5 and 6 equal to plus 1 and the ace and the 10 cards equal to minus 1. This system has better betting efficiency but lower playing efficiency.

If you keep a side count of aces (which I don't recommend for a novice counter—you have enough to worry about as it is), these systems are of equal value. The best way to play the modern game of four to eight decks is to "back count," that is, to watch the game from behind the table without playing until the deck is to the player's advantage. Then, you sit down and make a bet without having to play at a disadvantage with the house. If you are going to do this it is best to use a system which has a high betting efficiency. You should use, in my opinion, either the HI-LOW, Wong's halves system, or my combination of these. Lines 510 to 550 in the program control the count calculation. Although the program does not give the option of changing the count you can change the HI-OPT I to display the HI-LOW by changing line 530 as follows: 530 IFD>1ANDD<7THENCQ=CQ+1ELSE IFD=1THENCQ=CQ-1. You can use my system by adding a new line as follows:535IFD=2ORD=7THENCQ=CQ+0.5 ELSEIFD=1THENCQ=CQ-1. Leave line 530 in the original form for this count system. Note that the program displays integer values of the counts. Don't change the sections of line 520 to 560 which are "R W" calculations because this calculates the HI-OPT II which the program uses to decide how the computer will play the hands the players do not play.

In any system, you get an advantage when you bet high on high counts (your best advantage) and low on minus counts (house advantage). This program uses a range of one to nine as the range of bets. While all other blackjack programs bet in dollars, think of these bets as bets in unit chips. There is never an excuse to bet a range of greater than one to nine (although the "back counting" method may be thought of as a bet of zero chips).

At first it was not clear how the computer was betting its player's hands. As it turns out, this program plays better than the scientific method of betting a fixed ratio of advantage to bet. It uses the much better concept of not increasing or decreasing the bets rapidly which hides the fact that the player is a counter. The house spots the counter (considered as undesirable) by looking for very rapid bet changes from a bettor who had made a series of small bets then makes a much bigger one. In Atlantic City, they cannot throw out counters at this time, but they can shuffle away a good deck

when they spot you. In Nevada, they can throw anyone out, so this form of hidden bet is even more important. You can normally get away with "back counting" unless you drop down a very large bet after watching a table for an extended period of time. Don't watch too long at any one table. For me, part of the fun of the game is to play the role of a fool at the table.

There are several other options available with this game. The percentage of each card value is shown at the bottom of the screen. Do not look at this when you are doing serious practice. There is no need to turn this off as it does not intrude as much as the display of the count does.

There also is the possibility of having the computer play all hands for an extended period of time. This shows what an expert player can do (but see below for two problems). The results of all plays can be dumped to the printer when this option is chosen. There also is a second screen available where the complete results since the game's beginning are shown. This shows the average count to date and the house wins, draws and losses. There also are two options not shown in the documentation. You can make backup copies of the program and any changes you made by typing (GOTO3000). There also is a way to save the results of a long session to tape by typing (GOTO1140), although how to do and use this is not explained in the instructions.

As with many programs, there are a few technical problems. The program allows and the computer will play cards to split aces. Although any pair can be split, nowhere but in Puerto Rico does the house allow you to play more than one card to each ace when they are split. The program gives the player an additional 0.1 percent with this rule. This program lets you play your hand before it checks for a dealer blackjack. If the house has blackjack, all bets made, including the doubles and split pair monies, are kept. This rule is only played in England. Although the play in Atlantic City appears to be the same, only the original monies are collected, and the double and split monies are returned. It is set back as if these bets were not made. This program gives the house a 0.1 percent additional advantage from this rule. The author of *Blackjaq* has informed me that he will be changing this in the new version of the game to the method used in Atlantic City.

One rule not offered to the player in *Blackjaq* is the insurance option. In this option, if the dealer has an ace up, the player may make a bet equal to the original bet that the dealer has blackjack. If the dealer does have blackjack, the player ends up even because the original bet is lost and the other bet is played at 2-1. The house is very generous to the player because this is normally a very bad "rip off" for the player and should not be taken. If the player is counting and, therefore, knows that the deck is more than one-third 10 cards, this is a good bet. Not having this bet available is a problem for the advanced player because the higher the count, the higher the bet and the higher the probability of the dealer having blackjack when the ace is up. This rule is so valuable to the house over the average player that almost all casinos offer it. The lack of this rule to the advanced player is about a 0.1 percent disadvantage.

The program *Blackjaq* is an excellent program which should be bought by all players who plan to go to a casino or who wish to know how we "counters" work our game.

(Federal Hill Software, 825 William Street, Baltimore, MD 21230, 16K Extended, $19.95 tape and $24.95 disk)

C.C.Dialer — Reach Out And Touch Someone

If there is any justice, there is a special place in heaven for those who have served as Telephone Chairman.

For those fortunate enough not to have served in this capacity, a Telephone Chairman (or Chairperson if you prefer) is the unlucky "volunteer" who has the responsibility to call everyone on the team, club, committee, etc., to remind them of the group's next activity and what and who to bring.

I've had this position of honor with Indian Guides, school groups and, of course, The Team. The next time I'm so bestowed, I'm prepared, because I now have *C.C.Dialer*. *C.C.Dialer* turns your 16K Extended BASIC 80C into the equivalent of an automatic dialer. It does this by generating telephone touch tones through your television speaker.

The program allows you to create a telephone directory of up to 126 names and numbers. This directory is then saved on tape and reloaded when you wish to use the "dialer." After loading the program, the first screen to appear is the Main Menu. At this point you have the choice of manually dialing a number using the numerical keys, or recalling the directory. The directory is really the useful part of the program. When called up with a single key command, 14 previously saved names and numbers appear for each page with up to nine pages per directory (126 names). Pages are changed using the left and right arrow keys; individual names and numbers on the page are chosen using the up and down arrows. When the desired listing is found, hitting [ENTER] "dials" the number.

What actually occurs is the TV speaker "beeps" and "boops" the same tones which would be generated by pushing the touch tone buttons for the chosen number.

To "acoustically couple" the 80C's tones to the outgoing line, you simply hold the telephone mouthpiece up to the TV speaker. *Crude*, but it works!

Chris Computers uses the CoCo's digital-to-analog converter to replicate the dual tones used by Ma Bell when a touch tone number is pushed. When the tones generated by the TV speaker are received by the phone company's equipment, the call is placed.

Because it uses tones, *C.C.Dialer* is compatible with the non-Bell long distance companies such as MCI and SPRINT. To use it for these services, one directory line is used to dial the access number. You then wait for the access tone, use another directory line for your access code, and a third line for the number being called. Actually, it's easier said than done.

The length of the tone is adjustable; I needed a longer tone when using *C.C.Dialer* with MCI. When stored as part of a telephone number, pauses can be inserted between digits. A "9" is often used in a business system to access an outside line. Using pauses, a stored number will "wait" for the outside dial tone and then continue to dial the number.

Any name or number in the directory can be changed at any time. The only enhancement I would like to see in the directory would be the ability to insert a number into an existing list and sort the directory either by Alpha names or numbers. However, it is easy to use as is.

For those of you with devious minds, *no*, *C.C.Dialer* does not generate "Blue Box" signal tones which allow pirating of long distance time. And since it does not connect to the telephone lines, no FCC registration is necessary. *C.C.Dialer* is also available on disk, with telephone number files stored on the disk for rapid access.

If you do a lot of calling to a lot of people at one time, *C.C.Dialer* will be of use to you, but if all you do is place a few calls at a time, you'd be better off letting your fingers do the dialing. *C.C.Dialer* won't replace my telephone dialer at work, because I don't do a lot of consecutive dialing, but it will be very useful when I'm leaving messages for those lists of people when I'm Telephone Chairman.

(Chris Computers, 6299 Alderwood Lane, Delta, B.C., Canada V4E 3E7, disk $34.95, tape $29.95. *User's telephone system needs Touch Tone compatibility*)

—Bruce Rothermel

A Serious Applications Program For Cassette

Label III (V. 1.1.) is a cassette-based program to generate mailing labels (or lists). A Color Computer with Extended BASIC is required. A 16K computer (the absolute minimum required by this program) will accomodate up to 50 records (a record comprises a first name, last name, address, city and state, ZIP code, and optional phone number); a 32K computer (recommended by the producer, Owl's Nest Software) will accomodate up to 200 records.

I am, by profession, a technical writer. So I naturally looked first at the instructions supplied with *Label III*. The five pages of instructions fully describe the program and detail operating procedures. Even a carefully-written procedure to produce a backup copy of the program (complicated somewhat because the program uses both machine language and BASIC instruction) is included. My *only* dissatisfaction with the instructions is that they lack an introductory section — an overview — describing briefly and nontechnically the program, memory requirements, and tape format. (Both the functional program and a nonfunctional source copy to facilitate program modification are furnished.) Note, however, that this information is included — it's just not organized as well as it might have been.

Now to describe the program. The program is comprised of a machine language "loader" to allow automatic program execution, a lengthy BASIC program, and a *fast* language subroutine that allows sorting on a maximum of three of the six fields in each record. The program is, to quote Owl's Nest Software, "error trapped. . . and prompting is very extensive." I agree. The program led me through every procedure (these are explained in the following paragraphs) without requiring a single reference to the instructions.

As I mentioned in the first paragraph, this program accomodates up to 200 records (50 if using a 16K machine), and each record consists of six fields:

1. First name
2. Last name
3. Address
4. City and state
5. ZIP code
6. Phone number (optional)

After typing "*CLOADM,*" the program automatically starts. The resulting menu offers eight procedures:

1. Input names
2. Delete names/clear memory
3. Locate names
4. Save records to tape
5. Load records from tape
6. List records on the screen
7. Print labels
8. Sort records

Input Names. If I select this procedure, I am asked if I wish to enter phone numbers in addition to names and addresses (I can omit the phone numbers later if I wish when using the print procedure). I am then asked for the data for each field (I use upper- and lowercase characters for best appearance). After the first address line, I can select an extra line to accomodate four-line addresses. After completing the record, the entire record is printed on the screen and a second menu gives me the options to continue to the next recored, to modify the one just entered, or to return to the main menu.

Delete Names/Clear Memory. This procedure allows me to delete a single record, or all records. If I choose the latter (to clear memory), the computer "beeps" and asks me to either verify my instruction or to return to the main menu.

Locate Names. This procedure allows me to go directly to a particular record without viewing the entire file. Again, a second menu gives me the choices to continue searching, to modify the located record, or to return to the main menu.

Save to Tape. This procedure allows me to name my record file, then save it to cassette. It automatically saves the file twice, creating a backup file to prevent the loss of data due to a tape defect. After saving the file twice, I can verify the recorded file or return to the main menu.

Load from Tape. This procedure loads the file whose name I specify from cassette and concludes with the main menu.

List to Screen. This procedure lists all file records — one record at a time — on the screen using my choice of five speeds. After listing the last record, I again get the main menu.

Print Labels. This procedure allows me to print my file either on labels or as a list on standard paper. I can print one, two, or three labels per line, and I can select how many sets I wish.

Sort. This procedure allows me to sort a file on one, two, or three fields — last name, ZIP code, and/or first name. This procedure utilizes a machine language subroutine and is really *fast*. Owl's Nest Software states that this subroutine will sort 200 records on a single field in less than three seconds, or on three fields in less than 18 seconds. When the sort is completed, I am given the main menu.

As you probably suspect, I was very impressed with this program. It is extremely easy to use and the instructions are complete. The program was carefully designed, as evidenced by the extensive use of prompts and menus and its error trapping. In these days when so much emphasis is given to disk drives and disk-based software, it is refreshing to find a serious applications program written for a cassette-based system.

(Owl's Nest Software, P.O. Box 579, Ooltewah, TN 37363, cassette, $19.95)

—Jerry Oefelein

Line Cross Referencer
A Lot Of Program For The Price

Utilities, utilities. I love utilities! I was quite pleased to receive a utility as my first software review from *the Rainbow*. This one is named *LNXREFR* (Line Cross Reference) and is published by Micrologic. And they pack a lot of programming into a program that costs only $7.95 (U.S.)!

LNXREFR is one of many utilities for the Color Computer (Extended or disk versions) put out by Micrologic. Among its features, it provides a sorted listing of all referenced line numbers from all occurrences of *THEN, ELSE, GOTO, GOSUB,* and *RUN*. It also allows for optional printer output and includes an optional REMark remover, which removes all unreferenced *REMs* and comments, and will remove the whole line number if it is only a *REM*. If a *REM* is referenced, it will remove all the comments.

LNXREFR also has a line protect feature, which allows you to protect up to 10 lines from being removed. One item of note here. I had saved a program in ASCII (which all programs used by *LNXREFR* have to be) which contained *REMs* in line numbers 0 through 6. The program did not touch line number 0. It was left intact. Well, perhaps everything should start at line number 1, but this was not mentioned in the instructions.

The instructions are well-written and take you through everything step by step. They include a section on problems and customizing.

One item I had to fix (because I have disk drives) was the high-speed pokes. The instructions were very specific. They led me to all the places and showed me all the changes that had to be made. They even show you how to increase the number of lines that can be processed by the program and how to recover from an error while running *LNXREFR*. An error can occur if there is not enough cleared space set aside, the *DIMmed* variable is not set high enough for your program, or there is a syntax error in the target program.

Among the plusses for this program, it is user-friendly, with audio used when necessary. The instructions are adequate. The program checks if a disk system is being used and modifies itself accordingly, using different prompts and messages. The program tells you at the end of its list of referenced lines how many total bytes (non-ASCII) and how many total lines in the program. A nice little added touch.

The print routine allows for a margin to be set for each printed line. I have used another one of their products called *LLSTFM* (*LLIST* Formatter). The print routines in this program were much more extensive. One feature *LLSTFM* had which is missing in *LNXREFR* is a user-selectable Baud rate. It was very disappointing to run through the program and try to print out the results only to realize that it was still sending out information at 600 Baud while my printer (the DMP-200) was set for 1200 Baud. So before you run the program, type *POKE* 150,41 to put your system into 1200 Baud and everything should be okay (600 Baud users need not worry about this).

One more note. Since the program is written in BASIC, modifying it to suit your needs or fix something as described earlier shouldn't pose too much of a problem. Just remember, as menioned in the instructions, since the program uses an ML routine built into it at the end of the BASIC program to disable the [BREAK] and [SHIFT][@], never save a copy of the program for running purposes in ASCII format, for this will eliminate the ML code.

Micrologic is putting out some good software values, especially considering the price. I use *LLSTFM* every time I list a program on the printer, and although I don't use a cross-reference program as often, I'm sure you will liken to Micrologic's *LNXREFR*.

(Micrologic, Box 193, First Ave., East Brady, PA 16028, (412) 526-5781, $7.95 on tape)

—Eldon Doucet

Pigskin Predictions Passes With Dazzling Effects

Predicting the future is rarely an exact science. More often than not, crystal balls, intuition and pure guesswork are involved. Not so with *Pigskin Predictions*. This menu-driven, multifaceted program rivals the flexibility of the object of its machinations—the National Football League (NFL). While some people will no doubt see in this program an opportunity to increase their odds of winning illegal bets, the program is an excellent analytical tool for those who suffer from an affliction described by many football widows (and, increasingly, widowers) as NFL mania. Rainbow Connection Software even includes a disclaimer that the program is designed "for your entertainment and reference only," and predicts that its use will make you as good as most experts and better than most amateurs at predicting the outcome of NFL games.

The disk version requires 32K Extended BASIC while the tape version includes both 16K and 32K Extended BASIC programs and a data tape for updating scores. The 32K "enhanced" version features *Rainbow-Writer* screen displays. This screen utility program has been cleverly used to create very attractive screen displays mixing several colors, character sets, true descenders and other tricks in a 64-column format.

You begin by loading the 1983 schedule and initial ratings. The documentation informs you that it will take you from one to two hours to input the current season schedule and last year's final standings. But this tedious task has been done for you on both the disk and tape versions. You just select "Load Data" from the main menu, "Load Data" from the submenu, and then enter zero as the week desired. This data is included on the program tape right after the program (to protect it from accidental erasure). If the season has already begun when you buy the program, just update it with the scores of all games played to date and you are up to date. Updating of scores is done any time after the Monday night game and takes only about 10 minutes. Updating is a very sensitive process because a mistake means that its effect will be reflected in every subsequent week's results. So check your work very carefully.

The real power of this program is its ability to generate seven printed reports that provide you with all the data any NFL nut could possibly expect from a prediction program. Any printer that will handle more than 72 columns will do (this includes Model 33 teletypes). In the 32K version, double wide characters are used to emphasize headings and other facts (assuming your printer can handle this). If a printout is desired, just follow your menu selection number with a "P" and everything that goes to the screen also goes to the printer. Printouts are available for:

1) schedule by week,
2) schedule by team,
3) ratings by week,
4) ratings by team,
5) predictions and results by week,
6) predictions and results by team, and
7) division races by week.

The predictive variables include the home field advantage (number of points scored at home compared to number scored on the road) and a "power rating" (points scored vs. points allowed against an "average" team). The predicted point spread is the result of three calculations: 1. the home field advantage is added to the power rating of the team, 2. the home field advantage is subtracted from the power rating of the visiting team, and 3. the difference is the point spread. The point spread is then shown next to the favored team on the "Predictions and Results" printouts. If the game has been played, the actual score is displayed and a "C" is indicated for each score that came within the predicted point spread. While I obviously haven't used the program for an entire season, the documentation includes samples for the 1981 season that indicate after the eighth week (Oct. 25) the program predicted 57 percent of the weekly games correctly and was batting (passing?) 61.4 percent for the season.

Pigskin Predictions is a good example of the dazzling effects that can be achieved by combining a good utility program with a good application program. The use of byte-mapped memory for data storage runs circles around array and string storage in terms of memory conservation and processing speed. The program should prove to be a boon to those who are interested in predicitng the outcome of NFL games.

(Rainbow Connection Software, 3514 6th Place NW, Rochester, MN 55901, $29.95 tape, $32.95 disk)

—A. Buddy Hogan

Grand Slam: Bam, More RAM

By Jim Reed

I keep thinking, "This is like a birthing." I don't know why; I've never been near a birthing. Yet, there is this keyed up feeling, as if something very irreversible is about to happen.

The kitchen is crowded with people eating, drinking, talking. There is an undeniable electric charge of excitement in the air. That, too, is a bit hard to fathom since we are hardly breaking new ground or exploring a new realm. But I feel it, nonetheless, even though the focus of all the attention is nothing more than a 32 to 64K upgrade. And a simple "F" board modification at that.

So, what is all the fuss? I mean, I've seen "F" board upgrades done in hallways between meetings. Well, this is *my* "F" board, *my* TDP-100 and *my* new 64K chips, and *I* am going to do it, or be laughed out of town in the process. Somehow, I have become the only CoCo owner I know who doesn't have 64K, and it is high time I got with it.

You see, I'd been waiting around, for the most part satisfied with 32K in my personal computer, to see what Radio Shack was going to unveil in the way of its 64K CoCo. My theory was I'd grab one of the new models and be the envy of my block. Well, once I saw the new 64K, I liked it, but I didn't fall head over heels in love.

As a matter of fact, I sort of like the so-called "Chicklet" keyboard — and my TDP is already in a handsome white case, with ventilation holes where they're needed. I use a typewriter keyboard more than many people and, quite frankly, I find the wobbly, square keys on the CoCo a welcome relief from the all-business keys on my high-priced, office typewriter. Besides, no one has explained to me yet what I'm supposed to do with *Art Gallery,* BASIC*Aid, Platinum Worksaver* and the handful of other software programs that use keyboard overlays. So, upgrade, here we come.

I called DSL Computer Products on Wednesday. Two days later, on Friday afternoon, the UPS man delivered my *Grand Slam.* If my being managing editor of *the Rainbow* influenced the delivery schedule, well, so much the better. We all need our perks, don't we.

Friday evening, the main event. Representatives of Springboard Computing and Megahard are on hand; which is to say Charlie and Dr. Doom are there to hold my hand. Jenni occasionally peeks into the kitchen where my TDP is the centerpiece of the table. Dorris goes out for pizza. Yessir, this is a full-fledged *event.*

On opening the package, the chips are passed around for inspection. Charlie and Doom, both veterans of at least one upgrade each, nod their approval. Then, a kibitzer who shall go unnamed announces that these are "good chips," a declaration that appears to be based solely on an eyeball evaluation. I decide this means that they made it intact from Dearborn without getting run over by a truck. More reassuring to me is the 11-page instruction manual, complete with seven illustrations. Give me the printed word and a few pictures and I'll do brain surgery — on my electronic brain, that is.

Armed with a screwdriver to fend off overeager helping hands that keep finding their way inside my CoCo, I lay the instruction booklet aside, to be consulted only if I run aground. By now, Charlie and Dr. Doom have each found a screwdriver. Doom had brought his from home, a magnetic one that proves to be very useful. Reportedly, DSL has an installation tool kit, but since they didn't get the required (but refundable) $15 tool deposit, I didn't get to use their toolkit. No matter. A screwdriver is about all I need; the *Grand Slam* is a solderless kit.

The trough around the wick of a very wide candle is used to store the seven screws that hold the top cover on. Charlie removes a screw. Doom removes two screws. I get to do one all by myself. Doom announces that the two shorter screws are for the holes in the front, and sneers when I suggest that maybe I already knew that.

There is nothing to removing the RF shield, except that my hands are too big to reach under the motherboard and bend open the tabs. I grab the screwdriver, but the consensus is that someone with smaller hands should take over since I would surely manage to screw up, left to my own devices. I am allowed to read the instructions while Charlie deftly bends the tabs. The instruction book even tells the order in which to straighten out the tabs. Nothing to it, really. I truly believe most anyone, experienced or not, could install the *Grand Slam* without a hitch. Still, it is comforting to have a couple of know-it-alls sitting there telling you which hand to hold the screwdriver in.

Since I don't have the "chip extractor" that comes in the official DSL $15 refundable deposit toolkit, I use the screwdriver to pry loose the old RAM chips: 16 chips, two in each of the eight sockets since I had a "piggyback" 32K conversion. These come out as a unit, the jumper wires holding

them firmly together. They now reside in one of my 50 or so "junk drawers," just in case I might need them in the middle of the night.

Removing the new 64K RAM chips one at a time from the foam padding they were shipped in, I insert each one in a socket, mindful that our technical editor has called me for no other purpose than to caution me to "keep the notches lined up." The instructions are also explicit on this subject, "The white dots on the chips should point toward the rear of the computer." A piece of cake. I even develop a noteworthy technique of lining the pins up on one side of the socket and then raking my fingernail against the other pins to seat them in sort of a sweeping motion.

Trouble! Three jumpers have to be moved from the 16K position to the 64K position and a fourth jumper (supplied in the *Grand Slam* kit) dropped in. Ordinarily, this is a simple push-pull operation, requiring maybe 30 seconds to do all four. Trouble is, one of the pins has some extra solder on it and the jumper won't slide into place. I decide to remove the solder and a chilling chatter fills what had become a hushed room. Everybody has his or her own idea of what I am going to burn out or blow up with the soldering iron.

Panic sets in among the unfaithful when I elect to skip the soldering iron and instead heat up the screwdriver over the gas range in order to remove the little drop of solder. What should be a 15-second chore takes me about two dozen passes between the gas range and the computer, each pass punctuated with such helpful advice as "Careful now," and "Maybe you'd better bare wire jump it," or "You're going to catch something on fire." Finally, enough solder is burned off to permit the jumper to grudgingly slide into place. A sigh of relief is had by all.

The most tedious part is removing the eight capacitors. You see, I have needlenose pliers and I also have cutters, but I don't have needlenose cutters like those in DSL's toolkit. Regular cutters make do — with a little bit of force. While I point out to all gathered that the instructions plainly say that you need only clip one side of the capacitors and simply leave them in place, the assembled advisory staff concludes that total removal will "be much cleaner." It takes about 20 more minutes, given the clumsy tools I have to work with.

If I had that to do over, I'd most certainly leave the capacitors in place. Personally, I think that gives a cleaner appearance plus, you don't have to turn the computer upside down and shake it to get rid of all the tiny pieces. Also, as the instructions wisely suggest, if you happen to cut the wrong capacitors — admittedly unlikely — with only one end cut, it would be a simple job to repair.

Almost done. Before replacing the cover, we power up and, sure enough, the machine still works. I guess they are "good chips" after all. Did anyone ever doubt it?

A successful birthing! My baby now has 64K. The *Grand Slam* from DSL works, even when installed by a committee. Someone notes the time; with timeout for pizza, and a running tutorial from Dr. Doom during time-in, we have completed the job in less time than it takes to watch "Roots." It is noted that everyone present has personally witnessed someone else do the entire upgrade in less than seven minutes, . . . in a subway, . . . going through a tunnel, . . . on an "E" board. I am unimpressed and totally happy. After all, when you're having fun, why rush?

(DSL Computer Products, P.O. Box 1113, Dearborn, MI 48121, $75)

Fundfile Is Not So Fundamental

Fundfile is advertised by Parsons Software as a portfolio and account management program for funds and stocks using the Color Computer. Designed for disk-based systems only, one can use it to keep track of up to 50 securities and up to 900 transactions on a single disk. Written in open BASIC, it requires a minimum of 16K and a single drive. A printer is highly desirable for finished reports, although the program does generate screen reports as well.

The *Fundfile* package consists of the following four programs and some demonstration files on a write-protected diskette:

- Fundfile, the main program, is used to access any of the other three programs, although they may be loaded in directly.
- Fund-Rec is the data entry and direct access storage routine.
- Fund-Scr permits formatted screen callup of specific or all reports.
- Fund-Rpt delivers hardcopy printouts. (According to the accompanying literature, Radio Shack's LP VII, DMP-100, or equivalent is required. Actually, I believe that any CoCo compatible printer will do because there are no special printer format requirements.)

First, the good news: the program works. I must admit that I had some reservations about my ability to get it to do so. You see, I'm one of those people who still thinks that money is safest under the mattress. So I couldn't come up with a lot of "empirical data" to apply to this. In other words, I just don't happen to have a portfolio of stocks and/or funds lying around. In fact, I don't have a portfolio of anything lying around! So, for several days I just played with the demonstration data, changing values and playing big-time stock market financial mogul. I suppose I really didn't mind getting rich and suddenly poor using only electric money, but what's caution anyway except a virtue of the poor.

(Referring to Parsons Software advertising in this and other magazines, I discovered that *Fundfile* has a counterpart program called *Fundgraf* which is designed to make investment and divestment decisions for you—at a higher price!)

We're finally used to CoCo programs that work by now. So let's be critical and talk about those things which make *Fundfile* stand out for a moment. You get what the ad promises: selected and general stylized screen and printer reports of asset value, realized and unrealized capital gains, dividend payments, sales and transactions.

"Okay, Mister Reviewer, now really tell me about it. Let me know about the documentation and whether I could use the program as soon as I rip open the package, and all that stuff. In short, should I buy it?"

Well, since you asked, I will. Parsons Software, at least in its marketing of this package, seems to fall into that group of software houses or vendors who haven't been getting the complete word. That word is even if you come up with the most dynamic software yet conceived, skimp on the documentation and you're in trouble. Why so much critical attention to documentation by reviewers? Because, without clear, logically arranged and precise instructions, the end-user (you) will have trouble implementing the complete package. Unless the program itself is self-documenting, well-laid-out how-to instructions are just as important as the program itself.

Fundfile's "documentation" consists of 11 front and back pages of typewritten material stapled in the upper lefthand corner. Three of those pages provide sample printed hardcopy reports. There is no index or table of contents. It is broken down into the following sections:

I. Equipment Requirements
II. General Directions and Descriptions
III. Demonstration of the *Fundfile* (c) Programs with the Sample Data
IV. How to Start Your Own Account.

Those are not broken down any further. The meat of the literature must be considered sections III and IV, yet they are the most disorganized. Section III, for example, tells you how to run the demo files using all four programs. The author uses 80 lines of type in one paragraph for instructions before taking a breath and starting a new paragraph! In the meantime, however, it seems like I got lost dozens of times. I tried using a finger to keep my place, but that proved fruitless and inconvenient. And that's just one example of the effects of "afterthought documentation."

Everything is cramped and hard to visually follow. Unprofessional. Too bad. And too bad for us, too.

Some of this same lack of attention to detail is seen in the program's screen text formatting. I couldn't gain an appreciation for the obvious lack of symmetrical centering and spacing. The same applies to the hardcopy printouts, but to a less degree.

In short, it's attention to detail and making product use a breeze for the user that makes the difference between a good piece of software and an otherwise great product.

This was my first exposure to Parsons Software, therefore, I can consider myself completely unbiased and objective as a reviewer. I was inclined to think that maybe I had received a premarket package. But I discovered that they have been advertising *Fundfile* since at least the May/June issue of *Rainbow*. Hopefully, they'll take this opportunity to rework the "documentation." Otherwise, on a scale of one to 10, I give them an overall rating of five.

Therefore, to finally answer your last, and perhaps most pressing question, to buy or not to buy is your decision alone. I wouldn't until the entire package is cleaned up. At today's prices, what's $30 for a specialized program? You answer that one.

(Parsons Software, Dept. A, 118 Woodshire Drive, Parkersburg, WV 26101, disk only $27.95+ $2 s & h)

—Ed Lowe

An 'Interesting' Financial Program

That's INTERESTing is a clever title of a progam from Custom Software Engineering, Inc., that handles several varieties of interest calculations: amortization tables, bond yield, present value and rate of return.

The opportunity to review a program that figures loan amortizations could not have come at a better time, since my wife and I have just purchased a house and are now shopping for financing. The fixed rates being offered in our area are running between 12½ and 13 percent plus three to three and a half points, but we can get a five-year variable for 11 ¼ percent plus two points. The only problem with the variable rate is that there's no limit on how high it can go after each five-year period. Am I willing to gamble that the high rates of the early '80s can never come back?

The amortization portion of *That's INTEREST-ing* is very versatile. You can select any number of payments per year. You can input a number of payments or a number of years, and the program will tell you the amount of each payment. Or you can select the amount of the payment, and the program will tell you how many payments it will take to amortize the loan—or that the payment amount entered is not enough to amortize the loan at the interest rate specified.

You can stop at any point along the way and change any of the values for the remaining balance—interest rate, amount of payment, amortization time, etc. Instead of equal payments, you can specify a constant amount to be applied to principal.

We fed in a variety of numbers—different rates, different starting balances, 15- and 30-year amortizations—and came up with a surprise (to me) result: If we borrow $80,000 at eleven and three-fourths percent with a 15-year amortization, then sell our condominium within a few months and use the proceeds to pay down the mortgage, we could pay the whole thing off in five years before the rate changes. That makes the variable rate look like the best buy.

For any portion of the schedule, you can print out each payment or request only yearly and cumulative totals. Specifying a starting month is optional, but if you don't enter one, you won't get the year-to-date figures.

In the bond yield portion of the program, for a bond that you are considering buying, you enter the current date, the current price of the bond expressed as a percentage of face value, the coupon rate, the maturity or call date, and the redemption price. The program will return the current yield to redemption, the latter expressed in both simple interest (annual rate) and compounded (effective annual rate) form.

The present value program really should have been called present and future value, since you will frequently need to get past a "present value" which may be of limited use in order to get to a more useful "future value." The instructions include a very good example showing how much your deposits to a nest egg (such as an IRA) will be worth, and how this future value can be amortized. The program will tell you how many withdrawals of a specified amount you can make, or will give the amount for a specified number of withdrawals. The only catch is, you will not get hard copy for these

future value data unless you have first opted to print out a detailed computation of "net present value" whether you need it or not.

Another example given deals with a prospective investment in machinery. You enter your projected costs (outflow) and revenues (inflow) and your desired rate of return. A negative present value indicates that the investment does not meet your requirement. (A zero or positive present value would indicate that the proposed investment would equal or exceed your requirement.)

The rate of return portion of the program could then be used to find out what the actual rate is. Inflow and outflow specifications can be retained and reused as needed in both the present value and rate of return calculations.

If you want to find out the rate of return, that's really all the program should give you. But it doesn't stop there—it insists on giving you a net present value which should be zero, but often appears as a miniscule fraction expressed in scientific notation. A rounding subroutine should have been included to prevent that.

There is an undocumented bug in the amortization portion of the program. The instructions state: "If you did not want printer output, the schedule will be displayed on the screen in three columns. The output to the screen when each payment is displayed is intentionally slowed down to give you a chance to read it. If you want faster output, you may press the F on the keyboard. You may return to the slower rate by pressing any key other than F. If you did not request a full schedule, the program will give you the option of continuing after it finishes the amount of the schedule you did request. The program will produce totals at each stopping point."

The problem is that if you want less than a full schedule, and you press the F to shift into fast mode, your chosen stopping point will sometimes be ignored and additional payments will scroll by until the loan is fully amortized. If that happens, and you then enter another loan and request printer output, the override remains in effect and a full schedule will be printed. Even breaking and running again seems to be no help. The only solution that seems to work is to enter another loan, select screen output, and shift back into the slower mode.

The instructions, though generally clear and well-written, should have been better organized. The first two pages (of seven) claim to be general instructions, but almost all of page two is devoted to how to enter specifications for present value or rate of return. Since none of the information is applicable to bond yield or amortization, it doesn't belong in a "general" section at all.

There is a slightly confusing prompt that occurs when you select hard copy of an amortization schedule. It says, "SET PRINTER TO TOP OF PAGE." I took that to mean that I should turn on the top-of-form switch on my Smith-Corona, but I found out by experimentation that such is not the case. In printing a multi-page schedule, the program employs a line-counting subroutine that automatically leaves a margin at the bottom of each page and the top of the next. With the top-of-form switch on, the second page started printing too far down on the paper.

(Custom Software Engineering, Inc., 807 Minutemen Causeway, Cocoa Beach, FL 32931, 16K ECB tape, $29.95 + $1 S/H)

—Neil Edward Parks

Super 'Color' Speller
A Good Proofreader

This is my first exposure to a "Speller" program. I don't know what I expected, but I would call the *Super 'Color' Speller* a proofreader. What it does, it does very well, typical of the quality of the company's *Super "Color" Writer.*

Super "Color" Speller will read a text file from disk and compare the words enclosed by spaces with a dictionary list of approximately 20,000 words. Any words which do not match the dictionary list will be displayed individually for you to take further action. In 41 words or less, that is what the *Super "Color" Speller* does. Yes, you still need your own dictionary.

The *Speller* is very easy for the first-time user. Program execution is well-prompted and the documentation is nearly exhaustive.

Speller seemed to work as well with my *Scriptsit* files as with the *SCW*. It would not read my *Telewriter-64* files but they were not saved with *Telewriter's* ASCII option. It appears that it will work with most ASCII files, although Softlaw does not advertise it to be universal.

The program is sold to support the *Super "Color" Writer*; however, to use *Speller*, you must power down to get out of the *SCW* program and then load *Speller*. It would be much more convenient to be able to call the *Speller* program from the command line of *SCW*, make the necessary correction, and then save your file.

To use *Super "Color" Speller*, you first load the autostart program and then you are prompted to load a file from disk. After the text file is in memory, you are prompted to set one of two parameter selections. Your choice is to select any group of characters enclosed by a space or C/R, or only words which are reasonable. Next, it will tell you the quantity of words in the file. The next prompt asks if you want to add any of the text words to the stock dictionary. This option requires another disk because the system is write-protected. This feature is "very nice," but the program is so easy to use, I don't think the extra disk handling is worth the effort.

There is a slight pause to process the data and then you are into the main element of the program. The words which do not match the dictionary list are printed to the screen, one at a time while you select one of the operational features.

With each word put up on the screen you may: 1) ignore the prompt and move to the next word, 2) mark this word incorrect, 3) mark all words, or 4) quit. Marking the word incorrect will lead you to two other options at the end of the file. You are prompted to write a new file with either the marked words or with corrected words. Either option is easy to use.

If you choose to correct the words before writing a new file, each word is presented with a portion of adjacent text so that you can determine how the word is used to be certain of the spelling. If you choose to write the file with the words marked (with three asterisks), you can quit and reload *SCW*, load the new file, then use the locate feature of *SCW* to find and correct the marked words.

Other features of *Super "Color" Speller* include the ability to produce a completely custom dictionary, produce hard copy of the incorrect words and segment text files that are larger than available memory. *Speller* will perform a "memory sense" and make adjustments for any size memory from 16 to 64K.

Super "Color" Speller is pretty bomb-proof and very easy for first-time use. It would be put to best use proofreading longer documents but, if it could be loaded from the text editor of *SCW*, it would be used even more frequently.

(Softlaw Corporation, 9072 Lyndale Ave. So., Minneapolis, MN 55420; $69.95 for disk only.)

—Ed Sehlhorst

We Hadn't The Remotest Idea Now We Have Five

Among the burning questions facing those in the CoCo World today is this: Do I, or don't I need a *Spectrum Remote Reset?*

To this pressing query, I can honestly answer: No, I don't need one. That's because after testing out a review sample, I called Bob Rosen and ordered four more.

You say you're surprised? Let me tell you, I'm probably even more surprised. *The Spectrum Remote Reset* is the perfect answer to a question I'd never thought to ask. In fact, when I first saw a *Spectrum Remote Reset,* I laughed and said something akin to, "What in the expletive is that?" Upon showing it around the office, I could find no one who had the slightest idea what it was.

In the ensuing weeks since being introduced to this new gadget, I have tried to think of something to liken it unto; I'm still at a loss. An aluminum backscratcher? A burglar's tool? A croupier's rake? Ronco's newest TV offer?

Whatever you compare the *Spectrum Remote Reset* to, I believe you'll agree that it works splendidly and should last a lifetime. For those of you who've somehow missed the full page ad, the *Spectrum Remote Reset* is a hardware device that lets you press the reset button from the front, a "warm restart" trigger that's right at your fingertips when you're using the keyboard. Not the top item on your want list? Well, read on.

While there are many who can immediately appreciate the benefit of having a *Remote Reset,* I believe most of us need a little reminder of what it can do. I'm come to think of it as a relief valve. You see, while the reset is within arm's reach — unless you have a really compact installation, in which case your need for the *Remote Reset* is readily apparent — there's something about reaching to the back that amounts to admitting defeat. Reaching to the back breaks your concentration and can wreck your whole attitude. But not if you have the *Remote Reset.*

Yes, you say, but *EXEC 40999* accomplishes the same thing as hitting the reset button. True enough, but it seems by the time I'm thinking [RESET] the keyboard is locked up anyway. With the *Remote Reset,* you can sort of "sneak in" a reset with your pinky without ever lifting your hands from the keyboard. Thus, the *Remote Reset* is sort of a composure preservation device — and I can use all of those I come across in this business. That's why we have five *Remote Resets* at *the Rainbow* now.

A brief physical description is in order. The *Remote Reset* is all aluminum and is essentially a long, sturdy slat with three bends in it, a pin to hold it in position on the underside of the computer and a small hollow reamed out in the back to make it fit snugly over the reset button. It installs readily. It comes off easily, too, but I predict that, once installed, it'll have a permanent home.

Bob Rosen will be happy to know that the Spectrum Remote Reset fits the new 64K Color Computer as well as the vintage CoCo. On the other hand, I had to do a bit of custom bending to get it to work on the TDP-100. Perhaps there should be a TDP-100 model of the *Remote Reset.*

If you're still not convinced and I can imagine some of you are saying, "I'll just make myself one out of scrap metal or coat hanger wire," let's try and recall how successful you were the last time you used a coat hanger to roast marshmallows. Maybe you could fashion a dandy imitation, but, with too vivid memories of creating crude candy dishes out of tin cans in junior high school metal shop, I'll gladly leave the engineering to someone else.

Yes, the *Remote Reset* is just a hunk of metal with no moving parts at all, but it works without a hitch, does exactly what it's supposed to every time, would be difficult to improve on, and couldn't be more durable. For a piece of hardware, that's a combination that's hard to beat.

(Spectrum Projects, 93-15 86th Drive, Woodhaven, NY 11421, $12.95 plus $2 S/H)

—Jim Reed

Atari Joysticks Outlast Chewbones

Our family "puppy" isn't three years old yet, but he weighs about 120 lbs. and has long black hair all over (his body and the house). Gonga is a Newfoundland, a dog that most people say looks like a black St. Bernard. As the youngest of our three canines, he gets into trouble most often. I don't know how many pairs of shoes my wife has had "Gonga-ed," but I've begun to suspect she pours butter over her old shoes at the beginning of each fashion season.

We even had a couch Gonga-ed; a total loss! When he bent a heavy-gauge metal steak plate in half with his teeth, I began to think nothing was safe.

Well, Gonga seems to have met his match — Atari Joysticks. I kid you not. No, I didn't cover them with butter and yell "take it," a command on which he will leap into the air and chomp onto a ball bat held well over my head. But, somehow, he got a crack at one of them.

DSL Computer Products had sent me a set of Atari sticks with the necessary plugs to fit a CoCo already built in, and I took them home to test them out. I noticed the teeth marks only after noting that one of the pair of sticks seemed to be, shall we say, a bit "freer" in movement than the other. Yes, one was still stiff with newness while the other had been Gonga-ed. For a giant breed whose teeth sink clear through every tin can he manages to steal, Gonga had scarcely dented the Atari handle. Yessir, a real, true-life testimonial that, in my book anyway, beats any test Sampsonite ever put their suitcases through. Atari joysticks are tough.

In addition to toughness, Atari sticks have a handle big enough to wrap your fist around — and, to me, that's important, too. I can never quite "get into" an arcade game when the joystick must be held between the thumb and forefinger; dainty, I'm not.

If you have some Atari sticks, you know what I mean. Maybe you even have one of the many Atari stick/CoCo interfaces available. If you don't have Atari sticks, DSL's Atari sticks have the plug built right into the cable, so you won't need to buy an extra interface.

For the uninitiated, Atari sticks are a breed apart. There is a major difference between your Radio Shack, standard equipment joysticks and the Atari type. That is, Radio Shack's sticks use a potentiometer, and the position of the cursor — your paddle, plane, "man" or whatever — depends on the resistance in the potentiometer. The resistance varies according to the position the joystick handle is placed in. With a Radio Shack stick you can place the cursor anywhere within the screen's playing field. Not so, with the Atari.

The Atari stick uses switches instead of a "pot," and this has both its good and its bad points — mostly good, but while Atari sticks give you greater control on many games, on others they are practically useless.

The best example I can think of to explain the difference is John Fraysee's *Solo Pool,* marketed by Tom Mix. In *Solo Pool,* a single pool ball is placed on the table and you have to position the stick (flashing cursor) behind the ball in order to shoot (fire button). That's simple enough with any potentiometer-type joysticks, but next to impossible with switch-type sticks, which don't have any in-between settings.

Picture the pool table with me. With a switch-type joystick, your stick (cursor) can be in the exact middle of the table — Atari sticks automatically return to center on release, often a definite plus — or positioned at any of the end or side pockets. Additionally, the cursor can be placed on either end rail halfway between the end pockets. Thus, there are *only nine places* on the table that the cursor can be held *still;* otherwise the cursor is moving.

Of the Atari's four switches, one takes you from the center of the screen to the top-middle position on the screen. Pulling back on the Atari stick causes that same switch to take you back to center, or from center to bottom-middle of the screen — no in-between. Likewise, pushing an Atari handle to the left takes you from dead center to extreme left — you can't stop along the way; it's either all the way left, all the way right, or dead center.

So, how do you get to the end pockets then? Well, using compass directions for orientation, if you push the handle in a northeast direction, both the "north" and the "east" switch are activated and the cursor moves all the way up and all the way right, coming to a rest in the upper right-hand corner of the screen, or that particular game's playing field. Similarly, you push in a southeasterly direction for extreme bottom right, southwest for bottom left, and northwest for top left. If you want to go north by northeast, you're out of luck. With *Solo Pool* and an Atari joystick, your cursor can only be placed in one of the nine aforementioned positions.

In that case, what happens if you need to be in a different place in order to line up your cue stick of a cursor to hit the ball into the pocket. Simple, most of the time you miss. Occasionally, you get lucky with a three-railer or some such silliness.

At the risk of over-explaining, I want to be sure that we all understand Atari sticks, because you do need some — they're not just another pair of joysticks, but rather a different concept in joysticks — and DSL has them tailor-made for the CoCo.

Now, which games work with Atari and which don't? Well, I was surprised as often as not. As I suspected, with Atari sticks you lose quite a bit of control on games such as *Colorpede* — nine places you can remain still, otherwise you're moving. Nonetheless, I found *Colorpede* enjoyable with Atari sticks. On the other hand, I expected trouble with Adventure International's *Fire Copter,* but was pleasantly surprised to find that Atari sticks seemed to give me even more control.

In general, on *Pac Man* derivatives and other maze games you usually get equal, and occasionally better, control with the Atari sticks.

Another surprise came with Computerware's *Morocco Gran Prix.* I was sure that with Atari sticks I'd be in the ditch most of the time, but apparently something in the software program gave me much better control with DSL's Atari sticks than with my official RSI (Radio Shack issue) sticks.

No, I haven't compiled a comprehensive list of which games are better on which type of joystick, but I do hope everyone gets the general idea, which is to say: It's not really a case of either/or, but thanks to DSL Computer Products, now you can have both — without the added cost of an interface. The only unanswered question concerns those canine toothmarks: Was Gonga biting down in earnest? Or, was he simply playing a round of *Ghost Gobbler?*

(DSL Computer Products, P.O. Box 1113, Dearborn, MI 48121, $14.95 per stick, or two for $28.)

—Jim Reed

Think Tic-Tac-Toe Is Simple? Think Again

I was both anxious and curious to try *3-D Tic-Tac-Toe* when I received it to review. I must say, after playing it, I was not disappointed.

3-D Tic-Tac-Toe is a game which you play against your CoCo (a very experienced player), and is played on four planes, or levels. Each plane is a four-by-four grid and, by placing markers on the grids, you must get four in a row in either direction, up, down or diagonally.

I received the game, on cassette, with one typewritten page of documentation, which was completely adequate. Entering *CLOADM*, the program loaded and automatically executed with no problems.

At the beginning of the game, there is one screen of instructions and you are asked if you wish to move first or if you wish your computer to move first. Upon answering you go to the game screen. Pressing the "S" key will label the planes, rows and columns for you, which will make the game much easier until you get used to entering your moves. You then enter your move by typing plane/row/column. The computer will place your marker and then, almost immediately, make its move. If you enter a wrong move, by pressing "T," you can take back your move.

The game continues until either you or your CoCo gets four markers in a row or the game ends in a draw. When one of you wins the computer will flash the markers that compose the win. Pressing any key will then begin a new game.

The game played very smoothly and had very clear and concise graphics. I was able to beat the computer about 25 percent of the time, usually doing better if I made the first move. It is not an easy game to master and it does make you use the old "gray matter." I recommend this game, highly, to anyone who likes games such as chess, backgammon, checkers or tic-tac-toe.

(Oelrich Publications, 4040 N. Nashville Ave., Chicago, IL 60634, $16.95 on cassette)

—James Stewart

Software Review

Deprec Simplifies Depreciation Calculations

Depreciation is an important accounting tool. It allows businesses to account for the declining value of their equipment, and plan for its eventual replacement. Depreciation expenses for business equipment are tax deductible, too. There are a number of ways to calculate depreciation, depending on the type of equipment involved, and the pattern of use. *Deprec*, from B. C. Engineering, can help you do most, but not all, of the necessary calculations.

The program calculates depreciation by the following methods: straight-line, production (based on hours worked or units produced), double declining balance, and sum-of-the-years digits. Unfortunately, it doesn't provide 150 percent or 125 percent declining balance calculations, which are used for some classes of property, and doesn't adjust the first year's depreciation for the number of months of use, as required for some depreciation methods.

The program is easy to use. You are asked for the required data, one item at a time, and the results are listed on the screen. There's a minor bug in the program. If you do several sum-of-the-years digits calculations in a row, the "Totals" column will not be correct. B. C. Engineering assures me that this bug is being corrected.

The documentation for *Deprec* is extremely brief. It consists of a single page, describing the inputs required. While I don't expect a textbook on accounting, some discussion of when to use the various methods would certainly be in order. You'll need some accounting knowledge, or an accounting book, to make use of this program.

Depreciation calculations are complicated, and *Deprec* can help you with them. It would help even more if the documentation were improved, and the additional calculations were included.

Deprec requires 16K ECB minimum memory.

(B.C. Engineering, P.O. Box 768, Manchester, MO 63011, $10.95)

— David Finkel

The Companion
A New ROM-Mate for CoCo

Since your CoCo now has a multi-user capability, the need for additional I/O ports is reality. I'm not saying that *The Companion* by BTI is the answer, but it certainly is a long step in the right direction.

The Companion is a CoCo bus expander that allows the user to plug up to five cartridges into the ROM port at the same time. Essentially this gives the user five ROM ports for anything, including disk controllers, ROMpacks, real-time clocks, serial and parallel ports. A real-time clock and parallel port are offered by BTI for use with the interface.

The Companion actually consists of a plug-in adapter for the ROM port containing bus buffers, a 12" connecting cable, and a 14" enclosure containing the expansion bus (which means it will operate on the new Color Computer II without difficulty — an important plus). The larger enclosure also contains a power supply and decoding logic to make the system operate. In addition to the five slots in the top for your peripheral devices, a few switches and lights are evident.

On the front-right of the enclosure is a power on/off rocker arm switch with a lighted handle. Gone are the days when you forget to turn everything off. It's too bad the unit can't sense whether the computer and disk drives are still on. Five red LEDs are accompanied by a red pushbutton to enable manual selection and indication of the slot, or port in use.

These LEDs, I think, are a major plus, too, — and an advantage over the Radio Shack expansion device. It is sometimes difficult to tell *which* slot is active with other such devices. *The Companion* leaves no doubt.

Two of the external switches deserve some special consideration. On the side of the unit is a switch marked *On* and *Off*. This is a memory decode switch that controlls how the cartridge select signal will function. It recognizes the fact that a 64K program must differentiate between ROM and RAM depending on the memory page. One setting is used for 64K programs and another for all others. A restart button is immediately adjacent that will simulate an actual "Power On Reset" if your computer should happen to lock-up.

The actual selection of the port in use is very uncomplicated. After power-up you simply push the select button on the front of the unit to select any of five devices. The corresponding LED will light, indicating which one is selected. Cartridges are activated by an *EXEC 49152* instruction from BASIC as they will not auto execute with *The Companion*. With a disk controller, you must first select the proper port and then depress the restart button on the side of the unit.

If you would rather type than press buttons, all ports are also software selectable. *POKE*65504,X where X is the port number will enable any slot. *POKE*65504,0 will disable all ports.

The instructions explain both the actual operation and the theory of operation in detail. Full schematics and component drawings are included. I would say the documentation is adequate. I like the idea of BTI providing parts lists for all components. For an investment of this nature, it is nice to know that parts are not proprietary. Enough documentation is provided that an experienced electronic technician could service the unit if necessary. BTI includes a 180-day full warranty on all parts and labor if you return the unit postpaid.

I was very impressed with the overall craftmanship of the unit. From an electronic view, the unit was of the highest quality. The circuit boards and components were very impressive.

I was, however, disappointed in the mechanical construction of *The Companion*. The plastic used for the case was flexible and loose-fitting. To point cut an example, the hole for the port select pushbutton was so sloppy it affected the operation. Sometimes it would work and sometimes it wouldn't. I guess cost constraints prohibited a moulded enclosure, but it sure would have prevented problems. At the same time, I'm sure the problem I encountered was caused during shipping due to the flexible nature of the enclosure.

Finally, as most of you are aware, Radio Shack is marketing an expansion interface for the CoCo. It depends on your point of view as to which is the better buy The Radio Shack unit is more compact, more attractive, has the same features and has a moulded plastic case matching the CoCo. *The Companion* has the LEDs, a flexible cable and one more part. Take your choice as to which is more important to you. BTI's unit does cost $25 more.

(Basic Technology, Dept, Q, P.O. Box 511, Ortonville, MI, 48462, (313) 627-6146, Model BT-2000, $200)

— Dan Downard

Amortization Is User Friendly and Useful

Amortization is defined as extinguishing a debt by means of a sinking fund. *Say What?* If I wrote out the amortization formula for figuring monthly payments *Rainbow's* printer's probably couldn't print it, you would experience agonizing difficulty if you ever used it, and then it would only give very limited information. It is that complex. The definition and formula aren't very useful, but *Amortization* is. Before I get into any specifics let me give you a little nice-to-know information about *Amortization.*

It is in Extended BASIC and requires 16K minimum to run. It actually is only 3.794K long but evidently uses up a lot of space in buffers since the user must do a *PCLEAR 1* before *CLOADING.* If anything is tricky about the program, the latter is it. *Amortization* is very straightforward and user friendly.

Amortization provides you with everything you ever wanted to know about any amortized loan. It will work with the screen or printer. For each month it gives you what you still owe, how much your total monthly payment is, how much of your monthly payment went to pay for the principal, how much of your monthly payment went to interest, how much interest you have paid to date, and what your total payments add up to so far. Anything else you can think of a loan?

I found the menu options to be complete and self-explanatory. If you can remember to *PCLEAR 1 before you CLOAD*, you can throw the instructions (clear, short, and comprehensive) away. The menu is that good. Once you insert the item, the total amount borrowed, the number of months, and the interest rate, you can go to the next consecutive month, jump to any desired month (forward or backward), or go the main menu where you can make changes or start over, any with the press of one key. The screen always keeps you aware of what the program is doing, and that is also a nice touch. You never have to worry about whether or not the program bombed.

As I said, the menu options were nice but what I liked best was the output to the printer. On each page of the printout it gives you a heading and puts the information in columnar form with six columns. When it's finished giving you the particulars about all the months, it gives you the total interest paid, total principal paid, and the total interest plus principal paid. A really nice touch, but scary if you are a penny-pincher like me. I would suggest you do a couple line feeds before you start the printing so you will have a nice top and bottom margin.

Even though *Amortization* is not written in machine language, I found it fast enough to suit may needs and I am pretty picky about speed. Because it is in BASIC, you can modify it, even though I can't think of any modifications needed, except possibly the printer Baud rate (which is now 600).

I think *Amortization* will prove beneficial to me and to anybody who has, or plans to have, amortized loans. These are among the kind of loans common on cars, homes, and home improvement. It would sure be nice if institutions which make loans sould give customers a printout such as *Amortization.*

(Harmonycs, P.O. Box 1573, Salt Lake City, UT 84110, $11.95)

— Herbert Ridge

Hint . . .

To Hex With Decimal

To convert Hex to decimal: ?&HXXXX

To convert decimal to Hex: ?HEX$(YYYY)

The four Xs equal the Hex number, while the four Ys equal the decimal number.

Disk Manager
A Take-Charge Utility

An attentive freshman business student knows that a good manager is one who plans, organizes and controls the work that needs to be accomplished. *Disk Manager* measures up on each count. *Disk Manager* is a disk utility program written in BASIC for either the 16K or the 32K CoCo. Both versions are supplied on the same cassette. The medium chosen is unusual considering the end use of the software. The 16K version actually uses two separate programs, but functions otherwise identically to the 32K version (of course, you sacrifice file space — 32K will catalog 230 files, 16K about 95).

Disk Manager is first saved to each disk in your collection. I've learned from the use of this program and others that it is a good idea to leave about 10 granules free on each disk for unforeseen uses. The program has two major sections: one for single disk management and one for management of your disk collection.

Single Disk Management

The program begins by prompting for today's date. The date can then be added to any existing files on the disk or it can be incorporated into any file when it is being created. Updating of files and disks is a breeze with this feature.

With one letter commmands you can do the following:

1. Print a directory including the date of creation, today's date and the normal information contained in the RSDOS *DIR* command,
2. Print a Super Directory that also includes the size of the file in sectors and bytes, the granule numbers in which the file is located and the Start, End and Exec locations for machine language files. This feature was meant to be used with a printer, so don't be confused by the screen appearance. The printout is advanced on the printer so that when it is torn off3. Toggle the printer on and off. When the printer is on, everything directed to the screen will also be printed simultaneously.
4. List all of the files on the disk. The files names are listed in two columns with ascending numbers next to them. You may run any program listed by just entering the number next to its name.
5. Create a table that shows which granules are in use on the disk. All 68 granules either show up as a pair of zeroes or their number is printed if they are used. This feature is helpful to the technically-oriented user.
6. Add dates to files that have been changed. The date is stored in an unused portion of the disk. At the same time the date is saved, a spare directory is also saved to an unused part of the disk. This feature allows for crashed disks to be rescued relatively easily.
7. Change the drive number that the program is addressing. This allows those with multiple drives to use all of the drives in sequence rather than constantly switching disks in only one drive.
8. Locate the Start, End and Exec addresses for machine language programs.
9. Rescue disks that have been clobbered by damage to the directory, or one that has a file on it that has been killed that you now want to restore. It is absolutely necessary that the user know what was on the original directory in order to use

this program to rescue the disk. Unless you are blessed with a photographic memory, it is essential that you make printed copies of the Super Directory after each change to the disk since the information contained in this printout is what is needed for the rescue mission on the directory and the killed file(s).

10. Print the menu for this portion of the program.
11. End the program.

Multiple Disk Management

By selecting *Catalog* from the main menu above, you may perform several functions to manage your entire disk collection. You may:

1. Add a disk's directory to the catalog of all disk directories.
2. Write the catalog to disk or read it from disk. This provides you with the titles of all the programs in your collection and the location (by title of disk) for easy access. This catalog may be sorted by a file name, extension, date, disk number or disk name. As the program is executing the sort, a binary countdown appears on the screen so that you can follow what is happening.
3. List to the screen or a printer, find a file in the catalog, delete a directory from the catalog, print the multiple disk management program menu and exit to the main menu.

The documentation for *Disk Manager* consists of six single-spaced, typewritten, 8½ x 11 sheets. In addition to an adequate explanation of the operation of the program, the documentation includes several useful tips for using the program to an even better advantage.

I wish the address and phone number for the supplier had been given so that I could have called or written for information or to have questions answered. *Disk Manager* is an excellent utility that every disk user should consider carefully. At $29.95 it offers all of the features found in some disk utilites that go for $99.95.

(Prickly-Pear Software, 9234 East 30th St., Tuscon, Ariz. 85710. $29.95)

—A. Buddy Hogan

Back Issue Availability

Back copies of many issues of *the RAINBOW* are still available.

All back issues sell for the single issue cover price—which is $2 for copies of Volume I, Numbers 1-8 (through February, 1982), $2.50 for Volume I, Numbers 9, 10 and 12 (through June except May, 1982) and $2.95 for Volume II, Numbers 9, 10, 11, 12 (March, April, June and July 1983). Also $3.95 for Volume II, Numbers 3 and 4 (October, November 1983). In addition, there is a $3.50 charge *per order* for postage and handling if sent by United Parcel Service and $6 for orders sent U.S. Mail. UPS *will not* deliver to a post office box or to another country. This charge applies whether you want one back issue or all of them.

Most back issues are available on white paper in a reprint form. Issues out of print include May, July, August, September, October, November, and December, 1982 and January, February, April, 1983. VISA, Master Card and American Express accepted. Kentucky residents please add 5 percent state sales tax.

Due to heavy demand, we suggest you order back issues you want now while supplies last.

In addition, copies of the cover *only* of the July, 1982, Anniversary Issue are available separately for $1 each, plus 50 cents shipping and handling. These are suitable for framing.

From A Galaxy Far, Far Away

I'm sure I was born a few centuries too soon. I am in love with science fiction, whether it be literature, films, or arcade and computer games. Of course, my favorite form of transportation is a high-powered spaceship (not your conventional "flying saucer") equipped with ammunition and thrusters to get me the heck out of Dodge, if I'm being attacked by enemy spacecraft. Alas, I found *the* game that puts me in seventh heaven . . . or somewhere in a galaxy far, far away, called *Guardian*, by Quasar Animations.

I was very impressed the first time I played *Guardian,* requiring 16K Extended Color BASIC, not only because of its similarity to *Defender*, one of my favorite arcade games (thank goodness, no more quarters), but because it provides more action than its counterpart.

Your spaceship has a very important mission: to boldly defend your planet from enemy spaceships such as the Lander, Mutant, Pod, Swarmers, Baiters, Pulsers, and Munchies which attempt to shoot you and steal your energy modules located at various points on the planet's surface. If a Lander succeeds in carrying an energy module to the top of the screen, the module will be destroyed and the Lander will become a Mutant Lander — a more powerful ship that chases you and shoots missiles. When all 10 modules have been destroyed, your planet's defensive force field will no longer exist and the planet will destruct.

Ah, but what defense capabilities does your fastest ship in the galaxy possess? Its movement is controlled by your right joystick and, very responsively, I might add. The up/down axis controls the vertical position of the ship, while the left/right axis is used to regulate the thrusters (speed) and reverse your ship's direction. This is great for getting you out of tough situations. If the joystick is positioned in the center, your ship will slow down and eventually stop — another great strategic quality.

The fire button controls your ship's main gun which shoots plasma bolts. These bolts are more effective in destroying your enemy at short range, and single shots are more effective than rapid fire. Because, although five plasma bolts may appear on the screen at one time, only the most recent bolt has deadly potency — the previous shots are harmless. Getting close to the enemy is a big challenge and very exciting (your heart rate increases significantly).

One of your greatest defenses is the Smart Bomb, controlled by your spacebar, which causes all enemy ships on the screen to explode. You can chalk up a number of points and get a lot of annoying aliens off your back. You are allowed three Smart Bombs per game (so use them wisely), but you can be awarded an additional one at the end of each completed wave. Helpful tip — If Swarmers are on the screen, don't use the Smart Bomb or you'll lose precious points.

If a Lander is destroyed while carrying an energy module, the module will fall to the planet's surface. If its fallen distance is greater than one-third of the screen's height, the module explodes on impact. So, you must catch the module in flight by touching it with your ship and bringing it safely to the ground. This takes tricky, fancy maneuvering.

The game consists of three plays, which are terminated when your ship is destroyed. One helpful hint I found that was not provided in the instructions, is to push the joystick up or down immediately after your ship's destruction, because if your ship is in the middle of the screen when a new play starts, your ship has a greater chance of being zapped by materializing enemy here than if it is at the top or bottom of the screen.

Guardian provides excellent Hi-Res graphics and astonishing sound. The screen not only provides the game play, but has a long-range scanner in the upper-central portion of the screen which displays you and your enemy, plus your current score is displayed in the upper-left section. One of the most impressive aspects of *Guardian* I found is the color and sound the thrusters made when the left/right axis of the joystick was pushed. This is not only an excellent example of how detailed *Guardian* is, but more importantly, a fine example of what a good Color Computer game is all about.

I recommend *Guardian* to anyone who enjoys a responsive, action-packed game, although I don't recommend the game for young children because they might find the action too difficult and fast.

Oh, and remember, as an old and wise man once said long ago, feel the force and be the guardian of your planet — just don't close your eyes like his apprentice did or you'll miss all the fun and action of this fantastic game.

(Quasar Animations, 1520 Pacific Beach Drive, San Diego, Calif. 92109, $27.95 tape, $29.95 disk)

—Susan Remini

Voice Pak Quality Speaks For Itself

By Bruce Rothermel

My CoCo just said the above title — really! Clearly, distinctly, and very understandably! This magical transformation was acomplished with the addition of Spectrum Projects *Voice Pak* - Voice Synthesizer. The Voice Pak consists of a ROM Pack containing a VOTRAX SC01 Phoneme synthesizer and a PIA (Peripheral Interface Adapter) chip with a pot to adjust pitch — and a data tape containing a series of programs allowing the Rom Pack to do its thing. 16K Programs are on one side — 32K on the other. Loading and using Voice Pak is very easy. First, like any ROM Pack, insert the pack while the computer is off, then power up and load the cassette. The first program, "Speak," is a BASIC program which, when run, loads and executes the machine language "translate" program which, is the computer/Pak/dictionary interface program. The first words will be spoken by your computer now as it announces it is now ready by saying, "READY" to you. You now are presented with a menu giving the following choices:

1. Direct input from keyboard.
2. Build a text from keyboard.
3. Speak text from tape file.
4. Return to BASIC.

After punching #1, you are in charge of what your CoCo says, and this is almost where this review stopped 'cause at this point, after typing a few test words and sentences, I lost control of the computer. First the kids invaded the sanctity of my office with questions like "what is that!" and "Who's talking?" and "Oh, Wow!" and shortly, "Move over!"

For the next four hours, my CoCo was talking in a slightly Swedish and East Coast U.S. accent saying things like: "E.T. Call home," "Bonnie is a Skunk," "U R A Q T" and even "Peter Piper picked a peck of pickled peppers." No amount of reasoning or pleading about the need to complete this review could pry them away. Even my wife, the computer widow, declared this a time of temporary truce and actually enjoyed the CoCo.

This is a great way to bring the family together even if you don't want to. In mode 1, you type and enter short phrases and sentences that will be spoken upon hitting the [ENTER] key. After the computer speaks out the phrase an arrow prompt appears for the next line of input. This mode is ideal for getting used to the sound of computer speech and for trying out new words. What makes the computer speech so understandable is a phoneme dictionary which is loaded into memory with the ML program. It contains about 200 words with their correct pronunciation stored. When a word on the dictionary is encountered, the accurate pronunciation is then recited; if the word is not in the dictionary, it reverts to the phonetic rules stored in the chip. This results in a very understandable output which is enhanced by your spelling some words as they sound rather than as they are spelled. Thus "Peter Piper Picked" becomes "Peter Pyper Pyked", etc. Words in the dictionary are virtually perfect; "One" is pronounced "one." You don't have to redo it as "une" to get a proper sound. If you use a word often, you have the option of adding this word to the dictionary with

the correct pronunciation. Thereafter it will be caught in the dictionary before reverting to the chips' pronunciation rules.

A pleasant surprise is that numbers and special characters are recognized by the ML program and will be spoken out. Numbers as high as 999,999,999,999,999.999 will be understood and spoken as trillions, billions, millions, thousands, hundreds, etc. The number 23 is pronounced twenty-three, not two-three. The phrase 100 x 3 = 300 is spoken as "One hundred times three equals three hundred." If you put a $ in front of a number the program assumes that you are talking about money. $123.45 is spoken as "One hundred twenty three dollars and forty-five cents."

Quotation marks are spoken as "Quote"and the # sign is pronounced as "number." Each comma, space and period produce a pause. Numbers can be pronounced individually by either putting a space between or by putting the "!" control code before the number.

The resulting statements contain inflection and human-like intonation in the sentences similar to what you would have in a normal conversation. I have no idea how these are added, but it's there.

Inflection can be changed via a pot on the ROM pack or through keyboard commands. Four speech tones are available. By changing tones you can have four different voices.

In no time at all, your CoCo is talking like your Uncle Bill — assuming your Uncle Bill was born in Sweden and is currently living in New York.

After beating the kids and wife away, I was able to advance to choice two of the Menu: Build text from keyboard. This mode allows you to input more than one line of text at a time. Using this, you can create long messages. Ever hear a computer say the Gettysburg Address? I have! You have the opportunity to save this text on tape or disk for later retrieval.

Choice three of the menu allows you to load and hear text which was saved in an ASCII file on tape or disk.

Choice four returns you to BASIC.

Disk users have to use Radio Shack's Multi-Pack Interface (#26-3024) to use both the disk controller and the ROM pack at the same time. The disk controller goes in slot four; the voice unit goes in slot three.

Making nifty sentences with your Color Computer is fun, but how do you use it for "practical"purposes? The Translate program can be easily merged with BASIC programs (or ML clear programs less easily) adding speech to programs. A typical addendum would be:

```
10 PCLEAR 1 (Clear video RAM unless needed.)
20 Clear 5000, &H5FFF (Reserve high RAM for ML)
30 CloadM "Translate" (load ML program stored at end of BASIC program)
50 DEFUSR1 = &H6005 (entry point for speech)
130 Phrase $ = "This is a Speech" (insert text)
340 X$ = USR1 (Phrase $)
```

Viola! Your BASIC Program now talks! Full details of merging the ML "Translate" program with basic programs to use the Voice Pak are included in the two operations manuals included with the package.

The dictionary section contains a dictionary manager program which allows you to:

— Add Words to the dictionary — with their desired phoneme pronunciation.
— Delete a word.
— List the dictionary and pronunciations to the screen or printer.

What are the uses for this program? Absolutely astounding! The uses are limited only by your imagination.

People with speech problems now have an inexpensive voice which allows them to talk to anyone on the phone. Conversations are as simple as typing in responses and hitting [ENTER].

Any programs can now be voice-prompting. Educational programs now can include pronunciation of spelling words. Foreign languages can be taught with the correct pronunciation. Adventure games can yell "Look out" and any other hints the author wishes to include. Arcade games can now include the voices of the control tower or you can hear the Aliens attacking.

Bob Rosen of Spectrum is out to make this the standard in CoCo Voice Paks. His chances of succeeding are high. First, the verbal results are Great. Second, the price is right. He is betting on low margins and high volume. Third, and most important, complementing programs are being developed by software houses. Computer Island, Prickly Pear, Jarb Software and Aardvark are in development or have released programs which utilize the Voice Pak. More will be coming.

I had the opportunity to run the French Foreign Language Program from Computer Island. It displays the French word — pronounces it — and then asks for the English translation. Congratulations are spoken for the correct answer; the correct word is given if the answer was incorrect.

Spectrum Projects is developing a smart terminal program using Voice Pak to announce the data received from a BBS or other information source.

This advancement, at this price ($69.95) is a breakthrough adding immeasurably to the utility of the Color Computer. Get One!

(Spectrum Projects, 93-15 86th Drive, Woodhaven N. Y. 11421. $69.95 ROM pack and Tape, plus $3.00 S/H. 16K min., or Dragon 64)

Hint . . .

Faster, Slower!

To speed up CPU, POKE&HFFD&,0

To slow down CPU, POKE&HFFD6,0

Elcircan
The Circuit Designer's Friend

Elcircan, from DLS Engineering, is an electronic network analysis program that is available on tape or disk. You may be asking yourself at this time "What the heck is network analysis?" Well, if you are involved in linear circuit design, either as a hobby or at work, this program will interest you because of its time saving potential. Linear circuits, such as audio amplifiers and active or passive filters, are usually designed to operate over a specific frequency range. To check out a prospective design, many calculations have to be made across intended frequency range to see if the circuit "fits" in the area of interest. This type of fine-tuning may require circuit value changes or design modifications to be implemented and then all the calculations have to be done again. This is where a network analysis program shines because it can predict the performance of a design by doing all these calculations for you at as many frequencies as you need. In effect, you can turn your CoCo into an electronic breadboard!

Elcircan models circuits using the nodal analysis technique. Nodes are connecting points between the various components in the circuit being tested. This method allows any configuration of circuitry to be modeled. The "answers" you get from *Elcircan* analysis for a given circuit is the ratio of output amplitude to input amplitude expressed in DB, the phase change between input and output signal, and delay in seconds which is a measure of how fast the phase angle is changing with frequency. The program allows you to use the following circuit elements: resistors, capacitors, operational amplifiers, FET transistors, inductors and NPN bipolar transistors. As with most circuit analysis programs, these components are assumed to be perfect, capacitors and inductors have no stray parasitic reactances, etc. These stray effects are important at high frequencies and can be simulated using nodal techniques. I should mention at this point that all network analyzers have various limitations, the computer term GIGO, (garbage in-garbage out) applies here also, a FET or transistor circuit should be properly biased and configured to be correctly analyzed.

If you have the cassette version and a 16K machine you have to do the *PCLEAR0* routine before loading. You are allowed to save or load circuits to and from tape and also circuit analysis in a like manner. This lets you build up a library of circuits and their respective operation for future use. A utility called LISTER is included and is used to list the saved output of *Elcircan* to screen or printer, it can also be used to list any file saved in the ASCII format.

The disk version of *Elcircan* includes a great utility that lets you use your printer to plot a display of frequency versus amplitude and phase, or amplitude and delay. Standard ASCII characters are used for plotting and should work on any standard ASCII printer. A hard copy of a circuit response is a valuable design tool and adds a professional touch to this program.

A review isn't complete without a little criticism; very little in this case. Being written in BASIC, large arrays may take more than a minute per frequency to analyze due to the large number of calculations to perform, however, you can dump data to a printer or file and review when the program is finished. Of course, changes are much easier to implement if you want to customize the program to specific requirements. A short routine to select Baud rate and 80- or 132-column format for instance. The manual is adequate but for the individual with little or no experience with this type of circuit analysis it may be somewhat brief. Several circuits using all the available circuit components and a step-by-step walk-through would be a big help here.

Elcircan is a very useful tool and should be added to the very long list of excellent CoCo software.

(DLS Engineering, P.O. Box 754, National City, CA 92050, $24.95 tape, $34.95 disk)

—Ron Hansen

Sonar Search Interesting Version Of Old Favorite

Do you remember, as a kid, using reams of paper and trees worth of pencils to play battleship? Though the details varied from neighborhood to neighborhood, the game consisted of two players, each equipped with pencil and paper, the paper being divided into an array of squares (16 x 16, 20 x 20, etc.), on which each player "hid" five war ships of varying size. The object was to "sink" all five of your opponent's ships by guessing their locations before he/she guessed yours.

One of the big toy companies has a fancy plastic board version and I remember seeing a battery operated version with lights and sound effects. I figured this had to be the *ultimate*.

Of course, the ultimate is never really the ultimate and that holds true in this case. Now, the old game is computerized! *Sonar Search,* written by Bob Schmerling, is an interesting variation of this game for the CoCo. Though not a machine language shoot-em-up, it is a pleasant change of pace, a *thinking* game.

The scenario is simple and straightforward. You're in a submarine equipped with sonar and a number of torpedos. The object is to seek out and destroy five enemy ships on the surface. What you actually see is a blue field of water covering about 80 percent of the screen and a small crosshair for aiming the sonar and torpedos. The bottom 20 percent of the screen is dedicated to designating players' turns and remaining shots.

You position the crosshair with the right joystick anywhere on the blue portion of the screen and press the fire button. If one of your opponent's ships is hidden at that particular location, a small colored box appears, indicating a hit but not total destruction. Moving the crosshair again, you try to figure out where the rest of the ship is hidden. Once you hit the ship enough times to sink it, a picture of the ship remains on the screen. The hidden ships require three to five hits depending on the ship's size before it is sunk.

The real challenge occurs after pushing the fire button and not hitting anything. A very realistic sonar sound is generated and by counting the number of beeps you get an indication of how far away the nearest ship is located but *not* its direction. You'll have to develop your own strategy in deciding which way to move next—up, down, left or right.

There are three modes of play, though the manual only mentions two. The first is the one player mode which is really a practice mode. You get an unlimited number of torpedos to sink the five hidden ships. This mode is where you develop the bloodhound instinct. After all five ships are sunk, you are told the number of shots fired, the number of actual hits and an accuracy score.

The second option is the two player mode—you against a friend. The scoring is similar to that of the practice mode, but with the added information of how many games each player won and all the other information for both players. The added incentive (or handicap) in this mode is the fact that each player starts with only seven torpedos for each round of the game. If one player manages to sink one of his opponent's ships, his opponent loses one torpedo on the next turn. The manual states: "In general, the number of chances that a particular player has at the beginning of his/her turn is determined by the formula N = 8 - (number of ships belonging to that particular player that have been sunk); N is the number of chances that the player has."

I discovered the third mode by accident. After loading and running the game from the first side of the cassette, I thought I would check to see if there were really three copies of the game, two on the front, one on the back. So I turned the cassette over and, lo and behold, the label on the back proclaimed there was a 32K version that would let me play CoCo. (All copies of *Sonar Search* both 16K ECB and 32K ECB versions.)

So I *CLOADed* this version and, sure enough, the menu had three choices: one player, two player and practice; the one player is you against CoCo.

CoCo doesn't play an unbeatable game (it doesn't cheat), but it does have the advantage of never forgetting its last position and the number of beeps it "heard." From one turn to the next there is no visual indication of your last location and everybody uses the same crosshair which always starts at the center of the screen. Remembering your last location is no mean feat considering the lack of land marks on the screen.

At this point, the two resident arcade experts were called in to give *Sonar Search* the once-over. The 12-year-old, Cheryl, said "Boring" and that was that. Tim, the nine-year-old with blisters on his joystick fingers, said "Neat" and has yet to beat CoCo. I believe this game may be too subtle for the hot-shot arcade set, but if you liked the paper and pencil version, this game is probably for you.

I say probably only because I got a very negative reaction to the price of $18.95. I'm not sure why, but that seemed high for a program written in BASIC. It is a well-written game and came over-packaged in a large vinyl-covered, plastic, clam-shell box which only contained the cassette and a small four-page instruction booklet. I personally have no objections to a plastic bag containing the cassette and instructions, especially if I could save a few bucks.

Sonar Search is written in BASIC for a 16K ECB CoCo and the TDP-100 but won't fit in a 16K Disk system. The 32K version needs (you guessed it) 32K ECB. The graphics are done in four-color medium resolution. The four-page booklet is adequate, although a bit confusing at times. The last note in the booklet suggests using a self-centering joystick for best results; I used both kinds and found both to be adequate.

As far as any recommendation for this game goes, I can only say that if you like battleship, and have run out of paper and pencil or people to play with, this version should please you. On the other hand, if you're into fast and furious shoot-em ups, you probably won't want this in your collection.

(S&S Arcade Supplies, 8301 Sarnow Dr., Orlando, FL 32807, $18.95 on tape)

—C.L. Pilipauskas

Hint . . .

Cold Poke

If for some reason you wish to simulate a cold startup, try the following: POKE 113,0:EXEC 40999. If you can think of any handy uses for this poke, drop us a line.

Reactoid: New ROM Pak From Radio Shack

Reactoid, a new ROM Pak game from Radio Shack, is a simulation of the operation of a nuclear power plant. The power plant is controlled automatically by a network of highly sophisticated computers, and all you have to do is to monitor their operation. The core of the reactor is surrounded by particle-emission tubes, that release energy particles which produce power when they strike the energy posts in the center of the core. The computer controls the "reflectoid," which diverts the energy particles towards the energy posts. Without the reflectoid, the energy particles would strike other particle-emission tubes and eventually cause the core of the reactor to suffer meltdown. As luck would have it, the computer system has just broken down. To make matters even worse, the primary and secondary backup systems have also just broken down. You are now on your own. You can either keep the power plant going manually, or fry in a meltdown.

When the game is started, the word REACTOID is formed letter by letter on the screen. The background color is black by default, but you can change it to either green or buff. The game is played with the left joystick, and it is started by pressing the fire button. The screen now shows you an overhead view of the reactor complex, with 24 energy posts in the center surrounded by 20 particle-emission tubes. At the top of the screen is the word MELTDOWN, with the first letter lit up.

The joystick moves the "reflectoid" around the screen, and the fire button is used to change the angle of deflection. As each energy particle is released from the particle-emission tubes, you have only one chance to position the reflectoid in order to divert it into the energy posts. When an energy particle strikes a post for the first time, it will light up, and you get 25 kilowatts (points). Each subsequent time that you hit it, you get another 10 kilowatts. Once all of the posts are lit, you then proceed to the next round.

The particle-emission tubes emit one energy particle at a time in round one, two at a time in round two, and so on. The scoring in subsequent rounds is multiplied by the round number, so it takes a while before you an really start to rack up the points. There are also bonus points awarded after each round for each letter in the word MELTDOWN that is not lit.

Of course, if the energy particles are not properly diverted, they can hit the particle-emission tubes, which will cause another letter in the word MELTDOWN to light up. After the entire word is lit up, you get to watch the reactor meltdown, and the game is over.

Reactoid is a fun game with good graphics and sound effects. The documentation provides a good background for the game, and clearly explains the game play and scoring. However, after playing it a while, I began to get bored because of its simple nature. As a result of this, I do not feel that it will hold the interest of the hard-core arcade buffs, who are used to more complex strategies. I do, however, feel that it would be an excellent game for younger children for the very same reason.

(Radio Shack, Catalog No. 26-3092, $19.95 ROM Pak)

—Gerry Schechter

Flexi Filer Has Professional Features

My first interest in microcomputers stemmed from the fact that I had a large phonograph record collection that was badly in need of organization. I was told that through the use of a database program I would be able to set up a catalog of my collection and sort and search for items at a rapid speed. For the past three years my data base software collection has grown at a rate faster than my record collection in search for the perfect Color Computer database program. I own or have tested just about everything from a $10 super simple system to the $200. *FLEX* systems. I now add to that experience a new package from Computerware called *Flexi Filer.* In a nutshell, this is an impressive and relatively inexpensive system that compares with some of the best.

First of all, the *Flexi Filer* has nothing to do with the *FLEX* operating system, and it can be used with a standard Radio Shack DOS. One disk drive will work fine with *Flexi Filer,* though it will support up to four drives. The multiple drive configuration simply gives you the advantage of more data space on a disk while you use the other drive for your system programs that are called by a series of menus. Happily, the program is not protected, which means that single drive owners may put the system information on their data disk and eliminate timely disk changing.

After typing *RUN "START"* the program shows a title page and the main menu. Defining a new file is very much like most sophisticated database programs and it is well described in the documentation. Fields are specified, labeled and saved in a file that is recalled when the database is run. You can define up to thirty-five elements (fields) per record and use up to 240 characters per record. *Flexi Filer* lets the user set up the screen display any way you want it on the standard 32 column CoCo screen. I would assume that setting up your own screen display for data entry makes for a more logical form when data is actually entered and recalled on the screen. *Flexi Filer* uses perhaps the easiest screen display format maker I have yet encountered in a database, and the user should find this an easy and fruitful experience.

Adding, deleting and changing records is likewise a simple and straightforward operation and shouldn't be a surprise to any first-time or experienced database users. There is one major flaw in this are that is perhaps the worst problem I encountered in the entire system. That is, if you type too fast you run the risk of skipping letters on the screen and your data entry is garbled. The only way to avoid this problem is to slow down and watch the screen for missed letters. As I was writing this review I received an upgraded version (3.1) of the program. I thought for sure that this problem would be fixed. This was not the case, however. Perhaps Computerware will be able to provide a fix in the future.

The number of records which may be stored in one database is a function of the size of each database record and what else you have on the disk. Computerware includes a one line program (PRINT FREE(0)*2300/(size) to help you calculate the approximate number of records your disk will hold. For example, if you have a one-drive system and a database of 100 characters, your database can have about 900 records. A multi-drive system that uses Drive #0 for the program and Drive #1 for data will yield about 1400 records,

100 characters long. This isn't bad for a program that uses a Radio Shack DOS that isn't noted for conservation of disk space.

Sorting and searching are also straightforward functions to use in *Flexi Filer*. As noted earlier, Computerware has upgraded the program and the sort section got the most improvement. All sorting is done via Select files. A Select file is a mini-file that sets up pointers to the main data file when screen or printed reports are called for in a given sequence. Earlier versions also had "disk sorts" where the entire data file disk is rearranged to a given sequence. "Disk sorts" are very time consuming and, in my experience with other database programs, often result in disk errors that can ruin your entire database. Owners of *Flexi Filer* should contact Computerware for upgrade information to Verison 3.1. The "select" functions provided with *Flexi Filer* are very complete and they should satisfy just about any general database user.

I left the description of the report writer functions till last as this area is perhaps the most complicated and unique part of the program. In general, *Flexi Filer* provides two methods of printing information—on address labels and on full sheets of paper. The formatting of both types of report are done on the screen through a rather unusual, but fairly easy to learn, screen editing program. Yes, *Flexi Filer* can cram a full size 80 column or 132 column report form on a standard 32 column Color Computer TV screen for editing. I found the use of the provided report worksheets a must, however, as the title area and records area of your intended report are shown on separate screens. I often print report data on tractor feed 3" X 5" cards and I found this format not supported by the stock version of *Flexi Filer*. Also not supported are arithmetic functions, except for the summation of all the numbers in a given field. On the plus side, you can save up to 10 different report formats for each database. These formats can be recalled easily by pressing a number from zero to nine.

Flexi Filer also includes an extensive disk information menu to help the user with disk functions. For example, you can list the directory or (on version 3.1) copy files from one drive to another. The thirty-two page documentation is compact, but complete and generally easy to follow. I would caution first time users to read the many appendixes included in the manual. Some of the information given in these "add on" pages should have been included in the main instruction portion. For example, only in Appendix "F" is there an explanation of why the screen format worksheets are included and that using them makes report formatting *much* easier.

In conclusion, this is an above average database program for the Color Computer. Users who do not have *FLEX* or *OS-9* will especially appreciate the availability of a professional filing system program, without going to the added expense of 64K and another operating system. My only major complaint is the speed of keyboard entry. Now it looks like I have no excuse not to return to that original task I purchased my computer for three years ago. The only problem now is that I'm more interested in reviewing software than cataloging phonograph records!

(Computerware, Box 668, 4403 Manchester Avenue, Suite 102, Encinitas, CA 92024, $64.95 32K Disk with Radio Shack DOS)

—Brian James

Quicksort: A Valuable Fast-Sort Utility

While BASIC is an easy language to learn and implement, there are some aspects of this programming language which are more involved than others, and it is sometimes easier to have a subroutine or utility handle these chores.

The sorting of string arrays, variable field lengths and numeric data seem to fall into these time consuming categories. Now, because of a new utility on our software market from Skyline Software called *Quicksort* or *Qsort,* this job has become easier.

Qsort itself is a high speed, machine language sorting utility that is made to be called from your BASIC program. It can also be incorporated into existing BASIC programs to improve their sorting speed.

To use *Qsort*, you must use the specific variables within your program as stated in the documentation. Strings are in a BASIC array called S$, with S$(0) being the first string. The number of strings that you want sorted would be contained in variable N. You must also define the starting position of the sort, the length of the sort field and a *DIMension* of at least 4.

All of the five variables must be included in your program otherwise *Qsort* will inform you of a "MISSING ARG" since they are essential for proper program operation.

Also included with *Qsort* is a test program called *Qtest.* This BASIC program generates random strings of random lengths and then calls *Qsort.* A typical sort of 100 random strings takes less than a half-second! It actually takes BASIC longer to generate these strings than it does for *Qsort* to arrange them in proper sequence. Wow! Sorting of longer strings is equally swift.

Besides there being a practical need for this type of program, and an apparent shortage of this kind of utility available, the time saved by implementing *Qsort* into your own program alone will justify the small cost of this fine utility.

Qsort is also compatible with your Disk system, since it loads at &HE00. It is also position independent.

(Skyline Software, 442 Sunnyside, Wheaton, IL 60187, $12.95 tape)

—Steve Schechter

The Gorilla Monitor

By Jim Reed

A number of years ago, when photography was my life, I read all the popular and professional photography magazines. When I got hold of a new issue, the first place I always looked was at the Spiratone ad. Spiratone, a New York mail order house, always had page after page after page of things to make a photographer's life easier. And, each month, there were always new items — a cross-screen filter, a clever new paper safe, this attachment, that accessory — all holding the promise of making it a bit easier to get that great picture we all aimed for. I was a Spiratone addict.

Photography is just an occasional, vocational necessity now, and I don't know if Spiratone is still around or not; now, I'm tuned in to Spectrum Projects for the same sort of reasons. It seems that if I've just acquired say, a widget, Bob Rosen at Spectrum is the advertiser most likely to have a widget-to-CoCo interface and the necessary cable to attach it to my whatsit. A deluxe model will have an LED and an extra switch that is "user-programmable." Now, I'm a Spectrum junkie. You can tell Rosen is both a "user" and a "hacker." As soon as you decide you need something, he already has it in stock waiting for your call.

This time he beat me to the punch entirely. Again, he sent me something before I even realized I needed it. Not that Bob Rosen invented green screen monitors, but he foresaw my need before I did.

Color Computer owners go through this cycle, you see. First, it's "I'll use the computer when the rest of the family isn't watching TV." Then, they say to themselves, "Well, I'll just take over the old TV in the bedroom all the time." Of course, to get the full-blown effect of a CoCo, eventually you decide "Let the kid pay his own way through college — it'll be better that way — I've got to have a brand new color TV set or I simply can't live another minute." And, in the back of your head there's this persistent, "Wonder what it'd be like to play *Startrench Warfare* on a giant projection screen?"

Bob Rosen doesn't stock giant projection TVs because he knows you really don't need one. He does stock the Gorilla 12" green screen high-resolution monitor because he knows you want one, or soon will.

Initially, I was pleased with the looks of the Gorilla monitor, but unimpressed with the resolution on the screen; that was because I hadn't yet changed the color signal to monochrome; that made a huge difference. Then I loaded a full-screen word processor. Wow! Even in the 85 x 24 high density display — used mainly for formatting a page — I could still read the display. On my 19" Sylvania Color TV, by way of contrast, in the 85 x 24 character set, even with the color turned all the way off and the sharpness control turned to maximum resolution, I still could only guess at some of the blurred words.

Now, I'm not saying I would want to write copy in the 85 x 24 mode, even with a Gorilla monitor, but I could if necessary. In the 64 columns by 24 lines mode, I could write all day with the Gorilla monitor, but I'd be blind if I tried that more than a couple of minutes with my big color TV. With the 51-column screen, both the green screen monitor and the color TV are easily readable. But the color TV's character set has halos while the Gorilla is markedly clear.

In a few weeks of use, I've found myself squinting and uncomfortable using the same color TV set I once thought was so terrifically sharp. I think I've developed another habit. The Gorilla green screen is so good, don't even try it once.

The truth is, before I compared the 12" Gorilla and my 19" Sylvania, I worried that maybe I was being unfair. After all, how could a little $100 monitor compare with my brand-name 19" pride and joy? Now that they're side by side —operating from the same signal thanks to Computerware's Video Plus interface — I keep reaching over to make absolutely sure the sharpness is at maximum on my Sylvania. It is, but it no longer "gets it" in comparison to the Gorilla. For one thing, on my color TV, when I use a word processing program the words tend to get "crunched up" if you're typing at the very bottom of the screen. Not so with the Gorilla, with this Hi-Res screen the words at the bottom are as clean and clear as those in the middle.

By the way, you do need some sort of video interface such as Video Plus to connect a composite video monitor such as the Gorilla to your CoCo. Naturally, Spectrum sells Video Plus, too.

I called on Dr. Doom to help me install the Video Plus. As is often his way, he took over completely and told me to shut up and take notes, "or whatever it is you writers do." Among Doom's comments on Video Plus: "This gizmo is a bit delicate. It could have been sturdier." After managing to pull one wire loose just inspecting the Video Plus board, Doom allowed as how, "*We* have to be real careful now." He did the obligatory lining up the notches, etc., diverting his attention momentarily to "straighten up your heat sink for you."

In order to have both a color signal for my TV and a monochrome signal for the Gorilla monitor, Doom had to add a switch to the Video Plus circuit. That went without a hitch, thanks to quite adequate instructions. Still, given a choice, I would have opted to simply pay a little more and get a switch installed by the elves at the factory.

No soldering is necessary with Video Plus — if you don't need both a color and a monochrome signal. We wanted both, so we located a U-shaped "wire loop" that Doom then snipped, soldering a single-pole, single-throw slide switch from Radio Shack across the cut ends. There was also a potentiometer that needed adjusting — easy enough given the good documentation — before we replaced the cover on the CoCo.

On replacing the cover, we faced a decision on whether to cut holes for the four new wires or just to route them out any available opening. We chose the latter, running the bigger wires out the video port and the thin wires out the 3-4 channel selection switch. Not the neatest installation, but practical.

The non-glare tube on the Gorilla does eliminate many reflections. Certainly, the green phosphor is easy on the eyes and graphics games, though not in color, do have a crisper display with the Gorilla monitor.

The Gorilla has only three control knobs on the front (on/off, brightness and contrast) and three more on the back (image width, image height and stablizer). By the way, the Gorilla does not have sound capability.

On a thorough reading of the manual, I find that there is indeed a tilt bracket on the bottom of the monitor to facilitate a laid-back viewing position. And in addition to several other telephone numbers for Leading Edge Products, Inc., which makes the Gorilla, there is a "service hot line" 800-number printed in red. I call it and get no answer, but then it is 3 a.m. I'm a night owl; this number must be for day owls.

As a last note, one of the showiest modes for the green screen monitor is with inverse video. This is not a built-in feature, but I pop in a software terminal program with this feature and it's very dramatic. I make a note to use it next time I want to impress someone. This Gorilla has found a home. As for Bob Rosen, I can see the handwriting on the wall; next month he'll be advertising a second keyboard/*OS-9* package so that I can timeshare with myself now that I have two screens.

(Spectrum Projects, 93-15 86th Drive, Woodhaven, NY 11421 $24.95 for Video Plus, $99.95 for the Gorilla Monitor, plus $5 S/H)

—Jim Reed

See you at RAINBOWfest

October 14—16 Fort Worth, Texas

As Games Go *Derby* Is Old Hat

Race cars traveling in a square maze eating dots and avoiding crashes with other cars make *Derby* very similar to the Atari game cartridge called *Dodge-Em*. In my opinion, *Derby*, is a below-average game. Written in machine language, it requires 16K and joysticks. Younger children may find *Derby* interesting, but it does not compare to the quality of games we have seen recently for the CoCo.

The scenario is as follows: you are in a speedy car and your prime objective is to run over all of the dots without crashing into a computer controlled car while moving through a mazelike track. The left joystick is used to switch lanes and the button is used to accelerate. It has much the same objective as the video game *PacMan*. Scoring goes as follows: you get 10 points for each dot you cross, and 100 points for clearing a whole screen of dots. You begin play with three cars and the game is over as soon as they are demolished. No high scores are recorded.

The instructors for the program are very skimpy and don't explain a lot of things that you need to know, such as how to use the joystick to control the car, or how many palyers can play at a time. The only colors on the screen are blue (maze), red (cars), and white (the background color). It appears that only one player is allowed; and one level of play is all that we found.

Derby will not go on my list of most wanted games. Better graphics and sound would have improved the action, but I think *PacMan*-like games are getting old.

(Bumblebee Software, P.O. Box 25427, Chicago, Ill. 60625, $14.00)

— Pat Downard

Submitting Material To the Rainbow

Contributions to *the RAINBOW* are welcome from everyone. We like to run a variety of programs which will be useful/helpful/fun for other CoCo owners.

Program submissions must be on tape or disk and it is best to make several saves, at least one of them in ASCII format. We're sorry, but we do not have time to key in programs. All programs should be supported by some editorial commentary, explaining how the program works. We're much more interested in how your submission works and runs than how you developed it. Programs should be learning experiences.

We do pay for submissions, based on a number of criteria. Those wishing remuneration should *so state* when making submissions.

For the benefit of those who wish more detailed information on making submissions, please send a SASE to: Submissions Editor, *the RAINBOW*, P.O. Box 209, Prospect, KY 40059. We will send you some more comprehensive guidelines.

Please do not submit programs or articles currently submitted to another publication.

Cosmic Clones Is Out Of This World

"They shoulda named it *Invaders Clones*," we thought as we played the first game, but come to think of it, the first 5000 hit rock 'n roll records all had three chords and a simple structure that consisted of first verse, second verse, chorus, third verse (which was usually the same as the first verse) and then a fade out. And we loved all of those records, didn't we? So why don't we just play this game and forget the intellectualizing?

We never got to answer that question because about that time a "Cosmic Clone" zapped us and the game was over. We only had 200-and-something points! Hey! If this thing's so formularized as to be boring, we ought to be able to stomp it into the ground in a couple of rounds, turn it off and forget it.

We never got to the turn-it-off part, either. Like a record with a good beat, *Cosmic Clones* hypnotized us. We fought the fiends late into the night. We came to know their every move. We grokked the clever way the Death Layer not only drops bombs and closes in on you like the spiked ceiling in a horror movie, but returns every shot you fire that misses a target.

The Clones divide into two bombs when hit, and you have to dematerialize both of them with you laser before they blow up your fuel cells, below. If a clone gets to the ground unheeded, it will zoom back up to the top and come down again as "Superklone." When a Superklone is hit it becomes two Clones, each of which becomes two bombs when hit.

The pace was picking up faster than the Death Layer was coming down. It was all too overwhelming — 'cause here we were — hooked again. Living proof that the magic of any good entertainment is in its ability to make you forget yourself and all your intellectual constructions and have a good time.

And a good time was had by all. This game is well done. (It even has a pause feature.) Oh, we sit around and make our snide remarks like everybody else. "The Death Layer should be graphically better." It sure is irritating to go to the title page so long between games," and like that. But all complaints were trivial in nature. (In fact, we have three other Mark Data products and find them all quite good.) Overall, the graphics and sound are excellent.

After playing for hours on end we still appreciate the subtleties of *Cosmic Clones*. There are no skill levels. None are needed. It starts easy, which makes it easy for anyone to play from the start and it gets hard which keeps all of us pros coming back again and again.

We are a father-and-son review team. So naturally Kat the Younger holds the family high score of 15,700. But he had help from above because at 10,000 points a Mother Ship started sneaking in and using a laser to materialize more fuel cells for him right before my very eyes. *Cosmic Clones* requires a minimum system of 16K Extended Color BASIC.

(Mark Data Products 4001 Alicia Pkwy No. 207, Mission Viejo, Calif., 92691 $24.95 Tape, $29.95 Disk)

— C.C. and Kat Courtney

Software Review

Colorzap — The Game

This program should not be confused with *Colorzap,* the utility sold by Software Options Inc. or with *Colorzap,* the disk utility published in *80-Micro.*

Colorzap, the game, is written in machine language to offer arcade speed even on a 16K BASIC machine. This is a very good game for computers without Extended BASIC. The feature I like most is, as you run up the score (bonus shields every 10,000 points) the skill level increases by two.

The game keeps track of the high scores and the level on which each score was made. *Colorzap* provides you with 15 skill levels to choose from. You are the large base in the center. Your mission, as described in the program, is to defend your "Stargate." The enemy ships appear in any one of four places, N, S, E and W on a compass. These are the four directions you can face using your joystick. The gauge on the left of the screen lets you know how many enemy ships remain during that particular wave. The gauge on the right tells the temperature of your laser. If you fire too rapidly for too long, you will overheat and lose one shield (shields are Spectral Associate's way of expressing lives in *Colorzap.*).

The different colored bars under the temperature gauge show how much shield energy you have remaining. When you complete a wave of enemy ships, something resembling a TIE fighter appears and it circles your base. If you don't hurry up and shoot it, it will fly into your base and you know what that means.

The *Rainbow Scoreboard* can start with my son's high score of 50,8000. He is our resident expert on game playing and he likes *Colorzap.*

(Spectral Associates, 141 Harvard Ave., Tacoma, WA 98446, 16K, $9.95 cassette, $14.95 disk)

—Ed Sehlhorst

The Other End Of The Phone

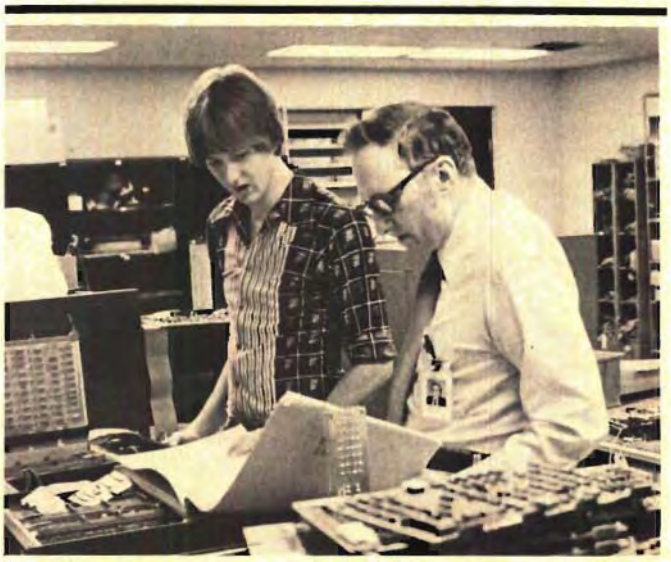
By Alexander B. Trevor

More and more CoCo owners are finding that their computer is not only useful and fun on its own, but with the addition of a modem and communications software it opens up a whole new world of data communications. Hundreds of computer bulletin boards (BBG's) around the country offer facilities for reading and posting messages, and often have free programs that can be downloaded to your CoCo if you have terminal software with "capture" capability.

Computer users from around the country tie into CompuServe's mainframe computers in Columbus, Ohio. CompuServe uses Digital Equipment corporation mainframes for extensive commercial data processing activities and for its Videotex service to home computer owners.

The commercial "information services" such as CompuServe, Dow Jones and the Source provide these and many other capabilities on a much larger scale. They are becoming increasingly popular, not only because of their useful databases, but because of their widespread availability via local telephone. In fact, using an information service is less expensive than using most "free" BBSs, except those in your local area. Long distance telephone rates are often higher than the charges made by these information services, particularly evenings, nights and weekends when the service's rates are as low as $5.50 per hour, but long distance rates are still $10-$18 per hour.

The information services are able to provide local access to their centralized "host" computers at rates less than long ces; CompuServe also operates its own national data network. The networks lease private lines from AT&T and other carriers most also operate some satellite channels to carry because they utilize nationwide public-packet, data traffic -data networks. The two largest packet networks, Tymnet and Telenet, are used by all three information servi-across the nation (and around the world). The lines are used to interconnect communications processors ("nodes") in each access city, and are operated at speeds from 9,600 Baud on voice grade circuits to 56,000 Baud on special high bandwidth circuits. The communications processors are minicomputers specially designed to handle data communications efficiently; all three networks have built their own special nodes.



CompuServe's communications processors, called MicroNodes, facilitate the flow of data throughout CompuServe's extensive telecommunications network.

Besides the "high speed" leased lines that connect it to the rest of the network, the node supports a bank of "low speed" modems connected to local telephone lines. These modems are set up as "answer" modems (to talk to your "originate" modem) and usually handle all Baud rates from 110 Baud through 1200. The Bell 103 modem standard is used by all the networks for 110 through 300 Baud; at 1200 Baud there are two standards: Vadic 3400, and Bell 212. Tymnet supports all three standards at all locations, but the other networks have many ports that support only the Bell standards, since they have become by far the most widely accepted.

When someone calls one of the ports on a network node, the node answers the telephone, brings up a carrier tone, and then listens for the corresponding "originate" tones. Usually, the modem listens for a short time for the 103 tones, and if those are not heard, it listens for one of the 1200 Baud

(Alexander Trevor is executive vice president of computer resources at CompuServe.

tones. If the caller is using 1200 Baud, then the modem modulation determines the Baud rate. But below 1200 Baud, it is more difficult to determine the Baud rate, since the same modem tones are used at all rates. The networks solve this problem by initially assuming a Baud rate for the first character, then examining the character(s) they actually receive.

Tymnet and Telenet require the user to type a carriage return [ENTER] in most cases as the first character; CompuServe uses control-C (or carriage return for network destinations other than CompServe hosts). This initial character will appear as different but distinct and recognizable characters depending upon the Baud rate of the originating terminal. The node then sets the port to the appropriate Baud rate, and prompts the user for whatever information is required to complete a connection to the desired host computer.

"When a user calls in to CompuServe from any location, through any of the networks, he may be connected initially to any one of 25 host computers."

By use of packet protocols, the networks are able to make very efficient use of their expensive leased lines. Most terminal sessions consist of considerable amounts of idle (or "think") time; during these periods space on the long lines may be used by other users, so no bandwidth is wasted. If bandwidth were strictly reserved for each user, as in simpler FDM (frequency division multiplexing) or TDM (time division multiplexing) schemes, then a 9600 Baud line could support only 32 300-Baud users or eight 12 Baud users. In practice, the networks' packet protocols support several times that many users.

The network connects the user to the destination host computer by means of a "virtual circuit" that is a logical path built from the originating node through as many as 10 or 20 nodes to the host adjacent node. The virtual circuit usually remains intact for the entire user session. Until recently, Telenet used a (theoretically) more reliable technique called "end-to-end" packet switching, in which every packet of data is separately routed through the network. This technique is supposed to make network link failure invisible to the user, but has the disadvantage of being much more expensive because of the additional overhead in routing each packet and the necessity of sorting packets at the destination.

Although the networks have all pretty much agreed on the best technique for data communications (with minor implementation differences), no such agreement has been reached on which host computer is best for an information service. Of course, nearly any computer can be used as a host — in fact, many CoCo's are used as BBS hosts. But even this giant of the micro world is far from having the capabilities required to support gigabyte databases (that's 1,000 megabytes, or over 6,300 Color Computer 156 kilobyte floppy disks) and the several hundred simultaneous users needing access to the data at any given instant. CompuServe uses Digital Equipment Corp. KL-10 processors in their own

"Vision-50" systems; Dow Jones uses IBM 4300 series computers; and the Source uses Prime computers.

In the case of CompuServe, each user is assigned to one of over a dozen disk "structures," each of which consists of one or more 200 megabyte disk packs. These disks live on different host computers from day to day, but all hosts always know where all the packs are residing. When a user calls in to CompuServe from any location, through any of the networks, he may be connected initially to any one of 25 host computers. When the user identifies himself by entering his "user id" number, then the host knows which host currently is the "home base"for this user.

A new virtual circuit is then built from the originating node to the proper host, which then prompts the user for his password. It is important to direct the user initially to his "home base" because that is where his own personal data files and E-mail messages reside.

During a session, a user may be "transported" to another host to gain access to a particular data base or service that resides on that particular system. For example, the "CB" simulator is a program that allows users from anywhere in the country to "talk" with each other via screen and keyboard on any one of 36 public channels. (Some apparently do more than talk: to wit, two on-line weddings that I know of, and countless "hot tub" parties.)

To permit any user to talk instantaneously to any other user with the required speed, their jobs must share a segment of memory, implying they must be on the same host computer. Another time that the user must be "moved" is when he accesses the Official Airline Guide (OAG). This data base actually resides on a computer in Chicago. In either case, a new virtual circuit is built through the network. Moving the user fom one host to another takes only a few seconds, and is infinitely faster and more efficient than trying to copy large data bases around to many hosts on demand. (The latter technique was employed by an early information service in England called Prestel; even Prestel has advanced to moving users between hosts — they call it "gatewaying."

The host computers used by the information services differ from your CoCo in a number of important respects besides price. The word length is larger (32 to 36 bits), the processors are faster and have more powerful instruction sets (and have more memory — up to 10 megabytes), and they have at least on high bandwidth path (or channel) to several strings of hard disk. Perhaps most important of all, they all have sophisticated operating system software, allowing the hardware resources to be efficiently used by many simultaneous users (or "timeshared").

The CoCo now has an operating system *(OS-9)* that has some of the features of a mainframe operating system, and, in fact, implements a two-user timesharing environment. However, I don't expect to see the information services running out to exchange their large machines for rooms full of CoCos. The reason is that in spite of their large price tags compared to micros, the mainframe computers are, still today, the least expensive way to provide high volume access to large data bases. As communicating micros become commonplace in homes and businesses, the information services and their mainframe computers will play an increasingly important role in providing easy and inexpensive access to elctronic mail, financial data, weather, special interest bulletin boards and many other services that have yet to be invented.

Launch Your Next Trip . . . From X-PAD

By Paul S. Hoffman

Now that vacation is over, we can start dreaming of that even greater automobile trip we'll take next summer. Or maybe you'd like to check what route will give you a certain amount of mileage for a running or cycling workout; or which route is the shortest between two points? Maybe you'd like to measure the course of a river for a canoe trip or for a geography essay. This program, *MAP-MINDER,* will measure any distance, no matter how circuitous or meandering, on any map (as long as your X-PAD Graphics Input Tablet can be slipped under the map).

First I'll give you the highlights of how to use the program, and then go into detail on how it works. You will need your X-PAD with either the supplied menu or a piece of paper with the menu area marked. Specifically, you'll need the 10 numbered spaces on the right side of the menu, and a line to indicate the edge of the left menu. Start by following the directions on the screen: locate the map's scale (a line in the area by the map title that shows how many miles or kilometers are in an inch or so of the map). Place the scale part of the map on top of the X-PAD, and press the pen down on one end of the line, then on the other end. The computer will calculate exactly how long (in inches) that line is. Then you will need to enter the length of the line in miles (or whatever units are indicated on the map scale). Digits are entered in sequence, using the boxes on the right side of the menu. The numbers will come up on the screen; when you've finished entering the number, press the pen in the left margin area of the pad to signal that you're through. (If you make a mistake, it's not much trouble to start over.) The computer will calculate and display the map scale in units per inch. Now lay your map route or part of it on the X-PAD. Start tracing the route with the pen (making sure you hold the pen straight up and down, and press just hard enough to have the internal switch make contact). The computer will display a constantly updated measurement of the total inches you've traced and what that means in map units. When you need to move the map, just pick up the pen from the surface, and CoCo will remember the accumulated measurements. Be sure to resume tracing in the same spot on the map after you've moved it.

(Paul S. Hoffman is an independent Color Computer programmer and designer for television and film. He is the author of Computerware's ***Semi-Draw*** *and forthcoming software packages for the X-PAD. Current projects include a sophisticated graphic animation system, including 3-D perspective displays.)*

How The Program Works

The 7" x 9" image area of the X-PAD represents 192 by 256 measurable picture elements. Each inch in the X-direction is exactly one-ninth of 256 pixels (HF=256/9) while each vertical inch is exactly one-seventh of 192 (VF=129/7). Each time two points are compared, their horizontal separation is figured (HS=ABS(X1-X)) in pixels and then converted to inches by dividing by the horizontal factor (HF). The same is done for the vertical separation (VS=ABS(Y1-Y)/VF). HS and VS are then used as the height and base of a right-angled triangle (the line in questions is the hypotenuse — the side opposite the 90-degree angle). The famous Pythagorean Theorem is used to calculate the length of the hypotenuse, which is simply added to all previously calculated distances.

Testing And Adjusting Your X-PAD

In a previous article on the X-PAD ("The Expressive, Expeditious, Exhilirating X-PAD," *the Rainbow,* January 1983) I complained about how touchy the Y-adjustment was on my unit. I've found it handy to keep a calibration routine at my immediate beck and call, so I frequently tack it on to

an X-PAD program. You'll find it here at lines 1000 to 1020. There are four adjustment screws under paper stickers under the upper right-hand corner of the X-PAD — that's the corner where the gray cord exits. The two holes along the top edge of the pad are for the X-adjustment, while the two along the side are for the Y-adjustment. In both cases, the screw closer to the corner controls the *centering* of that coordinate, while the screw away from the corner controls the *width*. Start by marking a piece of paper in the center (4¼" from each side in the narrow dimension, 5½" from each end the other direction). When you press the pen on this center mark, your data should be within two points either direction of 128(X), 96(Y). If it's not close, adjust the appropriate screw with a small flat-bladed screwdriver. After making sure the center is relatively accurate, check the X or Y width by placing the pen against the menu. On the left, the X-value should be within two points of zero (two points below zero is 254) while on the right it should be close to 255 (moving to zero and above if it exceeds 255). The Y-value should go from zero at the top (255 or lower if it's less than zero) to 191 at the bottom. To increase or decrease the range in either direction, adjust the screw further from the corner.

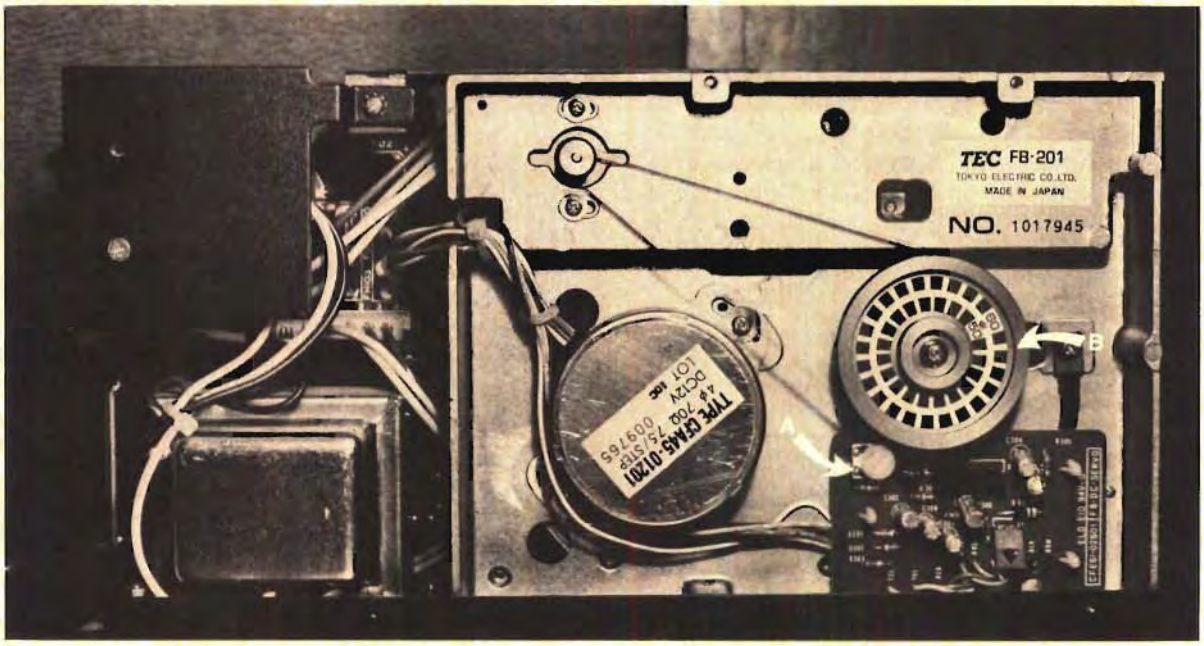
```
125.... 01FD
310.... 052A
END....0748
```

The listing:

```
10 '*********MAP-MINDER**********
20 '**REQURES EXTENDED BASIC***
30 '********AND X-PAD***********
40 '**PAUL S. HOFFMAN - 9/9/83*
60 HF=256/9:VF=192/7
70 CLS:PRINT:PRINT" PLACE MAP SC
ALE ON XPAD SURFACE -- THEN PRES
S THE PEN DOWN AT   ONE END OF T
HE SCALE."
80 GOSUB500:'GETS PAD DATA
90 IF(S AND1)<>1THEN80
100 X1=X:Y1=Y
110 PRINT:PRINT"GOOD!  NOW PRESS
 THE PEN DOWN    ON THE OTHER EN
D OF THE SCALE."
112 GOSUB510:IF(S AND1)=1THEN112
115 GOSUB500:IF(S AND1)<>1THEN11
5
120 GOSUB600:'FIND LINE LENGTH
125 PRINT:PRINT"THE SCALE WAS";L
;"INCHES":PRINT"LONG.
130 PRINT:PRINT" ENTER ON THE RI
GHT MENU MARGIN  THE NUMBER OF U
NITS (MILES?)    REPRESENTED BY
THE LINE YOU     JUST MARKED. (P
RESS THE PEN IN  THE LEFT MARGIN
 WHEN YOU ARE    DONE ENTERING D
IGITS.)"
140 GOSUB700:'READ RIGHT MENU DI
GITS, CREATING A STRING.
150 M=VAL(I$):SC=M/L:'SCALE=MAP
UNITS PER INCH
155 PRINT:PRINT"THAT MAKES THE M
AP SCALE EQUAL ";SC;"UNITS PER I
NCH.
160 PRINT:PRINT"NOW TRACE THE RO
UTE OR WHATEVER ON THE MAP - THE
 TOTAL DISTANCE WILL BE CONTINUA
LLY DISPLAYED.
170 GOSUB500:IF(S AND1)<>1THEN17
0
180 GOSUB510:IFS<>3THEN170
190 X1=X:Y1=Y:GOSUB500
195 IFS<>3THEN170
200 IFABS(X1-X)>250THEN300ELSEIF
ABS(Y1-Y)>186THEN300
210 GOSUB600
215 LL=LL+L
220 D=L*SC:DD=DD+D
225 CLS:PRINT@5*32+6,LL;"INCHES
230 PRINT@7*32+10,DD
240 GOTO190
250 END
300 PRINT@9*32,"  OOOPS! YOU WEN
T OVER INTO THE   MENU AREA - TA
KE A REST AND     READJUST THE M
AP.  THEN CON-    TINUE TRACING.
310 SOUND5,5
320 GOTO170
500 X=PEEK(65376):Y=PEEK(65377)
510 S=PEEK(65378):RETURN
600 HS=ABS(X1-X)/HF:VS=ABS(Y1-Y)
/VF
610 L=SQR(HS^2+VS^2):'MEASURES L
ENGTH IN INCHES
620 RETURN
700 I$=""
710 GOSUB500
720 IFS<>7THEN710
725 GOSUB510:IF(S AND1)=1THEN725
730 IFX>224THENRETURN:' USE LEFT
 MARGIN TO EXIT THIS LOOP
740 RM=INT(Y/19.2)+1
750 ONRM GOSUB770,771,772,773,77
4,775,776,777,778,779
755 PRINT@14*32+24,I$
760 GOTO710
770 I$=I$+"0":RETURN
771 I$=I$+"1":RETURN
772 I$=I$+"2":RETURN
773 I$=I$+"3":RETURN
774 I$=I$+"4":RETURN
775 I$=I$+"5":RETURN
776 I$=I$+"6":RETURN
777 I$=I$+"7":RETURN
778 I$=I$+"8":RETURN
779 I$=I$+"9":RETURN
1000 GOSUB500
1010 PRINTX;Y;S
1020 GOTO1000
```

Calibrating Your Disk

To calibrate your disk drive, adjust the potentiometer (A) while the motor is running until the spinning register marks (B) appear to stand still under a fluorescent lamp. Notice the separate scales for 50 and 60 Hertz current.



Install Your Own Disk Drive "On/Off" Indicator

By Richard S. Ellis

Here is a quick and easy modification for the TRS-80C Disk Drive that will help prevent leaving the drive on accidentally. Those with Disk Drives know that there is no "ON" indicator on the drives. With a little cash ($2 max), you can add a "POWER ON" LED to the front of the drive.

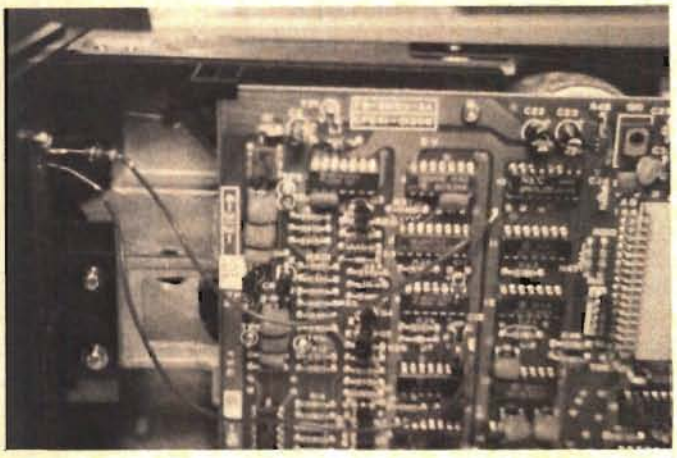
You don't have to be an expert at soldering, nor do you have to be an expert at drilling, so even if you have no experience, you should be able to put this modification in. You will need the following parts: one 5" red wire, one 5" black wire, one 330 to 800 ohm resistor, one LED (preferably large yellow or green), solder and superglue. If you have some clear fishtank air tubing it will make a more expert installation, otherwise, you will need some electrical tape. For tools, you will need a soldering iron and a drill.

Here are the instructions:

1) Turn off and unplug the drive. Take out the two screws on each side of the drive and then carefully lift the cover straight up. Set the cover off someplace, as you won't need it for this mod.

2) Look on the right side of the drive and you will see the component side of a large circuit board. Refer to the attached picture and find the 5V strip. Solder a 5" red wire (prestripped on both ends) to the point indicated on the picture. You will find an unused hole in the board there but it may be filled with solder. Move the free end of this wire out of the way towards the back of the drive.

4) On the right front of the drive above the door lever there



is a space about 2" by 3". Look on the inside and you will see a recession in the plastic where the wall is thin. Pick a spot in the recession where you want your LED, then drill a hole the size of your particular LED. Use the LED and make sure that it will fit the way you want. I had a bar type LED and had to use a small flat file to make the hole the proper size. When you get the hole the way you want it and have tested to make sure the LED will fit, proceed to the next step.

5) Clip the short end of the LED and both ends of the resistor to about 1".

6) Solder the resistor to the short lead of the LED.

7) If you have the air tubing, slide a 4" piece over the red wire, then solder the red wire to the other end of the resistor and slide the tubing down over the LED lead.

8) Clip the other lead of the LED to about 1".

9) Again, use the tubing if you have it and slide it over the black wire, then solder the black wire to the free end of the LED. Slide the tubing over the LED lead.

10) If you didn't have the tubing, wrap the bare leads with electrical tape.

11) Now lay the assembly down, plug in your drive and turn it on. Did the LED light up? If not, review the steps above and check your LED. If it did, turn off the drive, unplug it, then proceed to the next step.

12) Mount your LED, then glue it in.

13) Replace your cover with the four screws.

14) You are finished.

(Richard S. Ellis, a senior morse intercept operator for the U.S. Navy, uses his computer for radio-teletype transmissions and word processing. He also has an amateur radio license (KC5XS).)

All's Not Lost-We've Still Got FUN

By D.S. Lewandowski
Rainbow Contributing Editor

It seems that the final part of *RAINTEXT* has visited regions unknown, while searching for it, we have decided not to leave you without a column for this month. With luck, we should have it for you next month. Here are a couple of programs that you may find interesting. Both are rather short, but the principles are different.

The first one we shall call *MIMIC*. Wouldn't it be handy if we could send a copy of anything typed on the screen to the printer? If you happened to be working on an Adventure, (Has anyone solved *Sir Randolf of the Moors*?), you would have hard copy to show how you finally won the game.

To perform this feat, what we have to do is direct all characters being sent to the screen to the printer as well. Sounds difficult! Not really, our favorite computer jumps to a subroutine before it prints anything to the screen. The pointer for this routine is located at $168. So, what we can do is intercept this pointer to send the character to the screen. We send the character to the printer, then put the routine back on track, which sends the character to the screen.

The actual program will take up only eight bytes of code! Sure is a shame to have to *CLEAR 8,32760* just to use this program. I wish we could find a place in memory that BASIC sort of ignores. Well, there is always the memory just below the screen. BASIC "almost" never uses that. So we shall put our little program there.

So let's *ORG*inate the program at $3E5, which is just below the screen. Then, we have to move our pointer at $168. Loading the X register with the contents of $168 and storing it at *POINT*, will cause our program to contain the location in which the *PRINT* routine would normally follow. Next, we have to substitute the address of our routine. To do this, we load the X register with the address of HERE, and store it at $168. So far so good.

Now each time a character is sent to the screen this routine will send it to the printer via the BSR $A2BF. Then it will jump to the address we stored in *POINT*, which is where it would go normally. Wait one minute! What about the BRN THERE? Oh, well BRN is a Branch Never, this line of code causes the program to never branch to THERE. Why do I wanna do that? Let's say you wish to turn this routine on and off. By simply typing POKE 1010,32 you will change the Branch Never to a Branch Always. This will, in effect, turn off the *MIMIC* routine. By typing *POKE 1010,33* you turn it back on.

Okay, but I don't have a printer, so what about me? I'm glad you asked. Listing 2 is what I call a *FUN* program. Are you bored with the normal method of clearing the screen. *CLS*, and bing, a blank screen. Let's look into that. If we have a blank screen, the screen is really full of $60s, which means anything else must equal something else.

There must be another way. What if we load the X register with the start of the screen ($400) and, using it as a pointer, each check screen location for $60? If the location equals $60, we leave it alone. If not we subtract one from the current value until it reaches $60. That is just what the program in listing 2 does. Now, about DLAY, if you allow this program to run at full tilt, it clears the screen about as fast as *CLS*. By adding DLAY it takes a little more time, but this way you can see what's happening. Have fun till next month.

Listing 1:

```
PAGE 0001 THE MICRO WORKS

0001 0E00                ORG   $3E5      BELOW SCREEN
0002 03E5 BE0168 START   LDX   $168      CURRENT LOC.
0003 03E8 BF03F8         STX   POINT     SAVE LOC.
0004 03EB BE03F2         LDX   #HERE     PATCH ADDR.
0005 03EE BF0168         STX   $168      PLACE IT
0006 03F1 39             RTS             BACK TO BASIC
0007 03F2 2103   HERE    BRN   THERE     ON/OFF SWITCH
0008 03F4 BDA2BF         JSR   $A2BF     ?#-2 CHAR.
0009 03F7 7E     THERE   FCB   $7E       JMP OP CODE
0010 03F8 0000   POINT   FDB   0         SPOT FOR LOC.
0011 03FA                END   START

NO ERRORS FOUND
```

Listing 2:

```
0E00                00100        ORG   $E00
0E00 BE  0400       00110 START  LDX   #$400    SCREEN START
0E03 34  04         00120        PSHS  B        B ON STACK
0E05 5F             00130        CLRB           ZERO B
0E06 5A             00140 DLAY   DECB           B=B-1
0E07 27  02         00150        BEQ   OK       LONG ENOUGH CONTINUE
0E09 20  FB         00160        BRA   DLAY
0E0B 35  04         00170 OK     PULS  B        GET B OFF STACK
0E0D 7C  0E22       00180        INC   VAR      PASS COUNTER
0E10 27  11         00190        BEQ   STOP     255 PASSES?
0E12 A6  84         00200 LOOP   LDA   ,X       GET SCREEN LOCATION DON'T INC X
0E14 81  60         00210        CMPA  #$60     DOES IT EQUAL $60
0E16 27  01         00220        BEQ   NEXT     IF SO PASS IT
0E18 4A             00230        DECA           NO, OK SUBTACT ONE
0E19 A7  80         00240 NEXT   STA   ,X+      PUT IT BACK AND INC X
0E1B 8C  05FF       00250        CMPX  #$5FF    END OF SCREEN?
0E1E 27  E0         00260        BEQ   START    DO IT AGAIN
0E20 20  F0         00270        BRA   LOOP     DO ENTIRE SCREEN
0E22     00         00280 VAR    FCB   0        PASS COUNTER
0E23 3F             00290 STOP   SWI            PUT AN RTS HERE FOR BASIC
         0E00       00300        END   START
00000 TOTAL ERRORS
```

Greetings!

Psst! Wanna try some state of the art art? Try this:

```
TO CIRCLE
  REPEAT 36 (FD 8 RT 10)
  END

TO FLY
  PU
  REPEAT 10 (CIRCLE RT 36 FD 60)
  END

TO NOSE
  REPEAT 12 (FD 4 RT 30)
  END

TO EYE
  REPEAT 12 (FD 5 RT 30)
  PU RT 90 FD 3 LT 90 PD
  REPEAT 12 (FD 3 RT 30)
  END

TO EAR
  RT 25 FD 30
  RT 130 FD 30
  END

TO DOG
  NOSE
  RT 20 FD 10
  RT 40 FD 40
  PU LT 150 FD 23 RT 90 PD
  EYE
  PU RT 90 FD 30 LT 90 PD
  EYE
  PU BK 33 LT 90 FD 38 PD
  RT 190 FD 5 LT 15 FD 10
  LT 15 FD 10
  RT 90 FD 3 RT 90
  FD 10 RT 15
  LT 85 FD 20
  REPEAT 5 (RT 36 FD 4)
  FD 16
  LT 90 FD 7
  RT 90 FD 2
  PU FD 15 PD
  LT 15 FD 15 RT 10 FD 15 RT 10 FD 10 LT 5
  EAR
  LT 65 FD 15 LT 90
  EAR
  FD 100
  PU RT 115 FD 115 RT 90 FD 40
  FLY
  END

TO KENNEL
  REPEAT 2 (DOG LT 90 FD 95 RT 90 PD)
  END
```

Wanna see another one? Try this:

```
TO CIRCLE2
  REPEAT 12 (FD 6 RT 30)
  END

TO CIRCLE3
  REPEAT 12 (FD 4 RT 30)
  END

TO CATEAR
  FD 17 RT 135 FD 11
  LT 90 FD 11
  RT 135 FD 17
  END

TO CATHEAD
  CIRCLE3
  CATEAR
  END

TO CATBODY
  PU LT 20 FD 10 PD
  CIRCLE2
  END

TO CATTAIL
  LT 20
  REPEAT 6 (FD 6 LT 30)
  END

TO CAT
  CATHEAD
  CATBODY
  CATTAIL
  END

TO POSITION
  PU BK 65 RT 90 FD 75 LT 90 PD
  END

TO CATCAT
  REPEAT 4 (CAT PU LT 50 FD 90 RT 90 PD)
  CAT
  END

TO COPYCAT
  POSITION
  CATCAT
  END

TO REPOSITION
  PU RT 130 FD 156 LT 90 FD 40 PD
  END

TO COPYCOPYCAT
  COPYCAT
  REPEAT 3 (REPOSITION CATCAT)
  END
```

```
TO CIRCLECAT
  PU BK 50 PD
  REPEAT 9 (CAT)
  END

TO CIRCLECAT2
  PU BK 60 PD
  REPEAT 10 (CAT RT 4)
  END
```

Was it good? Did I make any mistakes? Did you figure out how to do it differently, or better? Try another:

```
TO CIRCLE4
  REPEAT 12 (FD 3 RT 30)
  END

TO SNAKEPOSITION
  PU RT 90 FD 10 LT 90 PD
  END

TO HALFCIRCLE
  REPEAT 6 (FD 4 RT 30)
  END

TO HALFCIRCLE2
  REPEAT 6 (FD 4 LT 30)
  END

TO BODY
  REPEAT 3 (HALFCIRCLE HALFCIRCLE2)
  END

TO SNAKE
  CIRCLE4
  SNAKEPOSITION
  BODY
  END

TO SNAKERETURN
  PU LT 90 FD 94 RT 90 PD
  END

TO SPINSNAKE
  REPEAT 10 (SNAKE SNAKERETURN LT 36)
  END

TO SPINSNAKE2
  SNAKE HOME LT 36
  SNAKE HOME LT 72
  SNAKE HOME LT 108
  SNAKE HOME LT 144
  SNAKE HOME LT 180
  SNAKE HOME LT 216
  SNAKE HOME LT 252
  SNAKE HOME LT 288
  SNAKE HOME LT 324
  SNAKE HOME LT 360
  END
```

Don't tell me! I know there is an easier way to do SPINSNAKE2. And I'll let you know what it is, just as soon as I figure it out myself. Now, why the results of SPINSNAKE and SPINSNAKE2 are not identical is beside me. Kris, I bet you know but aren't saying.

I have just started learning how to work the little eyebrow in the EDIT room. I find it a bit confusing, and wonder if anyone has any good ideas about exercises for working the eyebrow.

Here is what happens. First, I'm in the BREAK room. I press E, and immediately I'm in the EDIT room. The little eyebrow hangs there in the bottom lefthand corner of the green screen.

I press the right arrow and the eyebrow moves one space to the right. I press it again, and nothing happens. That's all the farther it'll go, I guess.

I press the left arrow and the eyebrow moves back to its corner. I press the up arrow and nothing happens. I press the down arrow and nothing happens.

Next, I press the spacebar, and discover that each time I press it the eyebrow moves over. I guess I'm placing blank spaces on the screen. I notice now that the little eyebrow will move to the right and to the left when I press the right arrow and the left arrow. Something may be happening when I press the up arrow and down arrow, but if it is, I can't tell.

Next I try placing the letter "O" several times on the screen. I notice that each time I type an O the eyebrow advances one space. I put a whole row of Os on the bottom of the screen. (If I reach the end, that line advances one, and the eyebrow goes back to the left bottom corner to start another row—but for now I want to experiment with this single row of Os.)

Okay, now I have some material on the screen to experiment with. I try pressing the left arrow and right arrow. The eyebrow moves in whatever direction the arrow points. That's neat, and it's also reasonable. Unfortunately, when I try the same thing with the up arrow and down arrow, what happens is NOT similar. When I press the up arrow, the line of Os goes up, *but the eyebrow remains in place* at the bottom of the screen. When I press the down arrow, the line of Os goes down, *but the eyebrow remains in place.*

In other words, the left arrow and right arrow move the

eyebrow relative to the text; but the up arrow and down arrow move the text relative to the eyebrow. That's confusing, and I think about writing to my Congressperson regarding this sad situation.

I have no trouble remembering that the right arrow and left arrow move the eyebrow, but I have had a lot of trouble remembering that the up arrow and down arrow move the image, until a friend showed me the following trick. You try it.

First, erase everything you've already typed. Next, type four spaces, then a single O in the fifth place.

Press RETURN, so that the single O on the fifth space advances up one. Then type three spaces, a single O in the fourth place, another space, and a second single O in place six. Then press RETURN.

Next, type two spaces, a single O in place three, three spaces, and a second single O in the seventh place. Press RETURN.

Next, type a single O in places two and eight; press RETURN. Then, a single O in one and nine; press RETURN. Keeping doing the one and nine for several more lines, say, 10 more. Then, on the 11th, type a whole row of Os from places one through nine.

Below that place nine "!"s (exclamation points). Do that for five lines.

Once you've done all that, press the CLEAR button and everything disappears, except perhaps for a single O at the bottom of the screen. Have you erased your wonderful creation? No. Hit the up arrow several times, and watch your rocket gradually rise across the screen. Hit the down arrow several times, and watch the rocket gradually return to its silo (under the bottom of the screen).

That's a little boring, but try this: press SHIFT and the up arrow at the same time, and the rocket pops up.

Wouldn't it be nice to have it fly up and away (past the top of the screen)? Why doesn't it? Try this: hit the RETURN key several times. Each time you do, you seem to be placing a blank space in the active text, and the rocket does slowly rise through the top of the screen.

Once you have the rocket all the way through the top of the screen, press CLEAR. Then press SHIFT and the up arrow at the same time, and the rocket emerges from its silo and flies off into the clouds. In other words, pressing SHIFT and the up arrow will scroll up whatever text is in the EDIT room—including blank lines.

Well, that was fun, and after doing it a few times I find that I have the ideas of up arrow, down arrow, right arrow and left arrow firmly lodged in the old *cerebris mediocus.*

What does SHIFT and the down arrow do? First, I tried it when the rocket was up in the clouds. Nothing. But I keep trying it. Nothing. So SHIFT and the down arrow is not the opposite of SHIFT up arrow. What *does* it do?

First, bring the rocket ship back down into its silo by pressing CLEAR. Next, raise it about six or seven lines so that it's halfway out. Now press SHIFT and the down arrow a few times. What happens? The bottom line on the screen disappears, and I wonder if SHIFT down arrow has just been erasing part of my rocket. I send it up to examine the damage, but am not certain what's happened. I do notice there is a gaping space in the middle of my rocket.

I press CLEAR again, and this time use the up arrow once or twice until just the tip of the rocket is sticking out. Now I press SHIFT and the down arrow three or four times. What happens? Well, the bottom line on the screen disappears. But the next line up (which contains a single O) remains exactly where it is. Does this mean that SHIFT down arrow has erased the bottom line? Am I gradually erasing the rocket ship, line by line, each time I press SHIFT and the down arrow? Let's see. Raise what's left of the rocket ship by pressing the up arrow a few times until something appears at the bottom of the screen. What finally appears? Voila! What appears is the missing line that you may have thought you erased.

Press SHIFT and up arrow once more, and see that the line under that is still there. You press SHIFT down arrow again, and a similar thing happens. That bottom line disappears. Press SHIFT and the down arrow about three more times. What do you see? Nothing? Right. Nothing. Whatever image is on the screen remains the same. So *what is going on*?

Well, it appears that every time I press SHIFT and the down arrow, the text is splitting downwards. I have not erased the rocket. Rather, I have elongated it: if I connect the lines with more Os, I'll have an ICBM. I have lots of good ideas, but I better go—I hear thunder. I remain,

Uncle Bert

(W. Bert Woofensburger ("Uncle Bert") manages his own hog and corn farm near Ypsilanti, Mich. He has recently acquired a Color Computer and is learning LOGO. Woofensburger's editor and assistant, Dale Peterson, writes for a living, and his recent books include "Genesis II: Creation and Recreation with Computers" and "Intelligent Schoolhouse: Readings On Computers and Learning." He is currently working on a book about Color LOGO with Don Inman and Ramon Zamora, to be published in 1984.)

HOME-GROWN HINTS AND EVIL-EYE

BY JOSEPH KOLAR
RAINBOW CONTRIBUTING EDITOR

It is hint time again! Newcomers to the Color Computer arrive unencumbered with preconceived notions. They must strike out on their own to solve many problems that crop up. Since there are no precedents for them to use as guides, they are apt to find startling but innovative solutions to problems that they had never encountered, much less imagined. These tailor-made solutions are quickly incorporated into their growing storehouse of knowledge and become a part of their operating system. The logic is that if these home-brewed solutions work for them, then they will work for someone else. That someone else could be you!

The old-timers, who once were newcomers, have their own tricks of the trade to make life a bit easier at the keyboard. Isn't it a shame that so many excellent tricks, solutions and hints lie dormant in somebody's computer room?

Let's make up a rule! A hint is anything that you discover that serves some purpose in making the use of the Color Computer simpler, saves time and ultimately, is fun. Don't sit on a hint because you feel it is unimportant. There is no such animal as a useless hint. It may be ignored as irrelevant to an individual's system, but it isn't useless. Many newcomers would like to know what you know. The rule is: pass it on through *the Rainbow*. Use simple language to explain the hint and help make life a bit easier for those who come after you to the wonderful world of the Color Computer.

Here are a pair of home-grown hints to give you the idea.

Do you use your printer frequently? When you detach the end strips from the fan-fold paper, do you toss them into the waste bin? Keep a cardboard box, or other container in a discreet location. Save the strips by dropping them into your special container. When loosely crumpled up, they make excellent packing excelsior (filler) when you mail parcels. They might make good kitty litter. What other uses can you dream up for the strips?

Do you use cassette index labels (RS 44-621) to affix over other cassette labels? Get double duty from the labels. If you use the index label, carefully peel it off the waxed paper backing so that the center strip remains attached to the waxed paper. Save this residue! That leftover strip will have a color-coded border. Use this handy bonus strip to label file folders or cassette storage boxes. Conversely, if you don't need the index label, you can still use the center strip. Just peel it off and use it. Put the index label back in the packet for future use. Now, what else can you use the "excess" strip for? Thank RS for the free, color-coded labels.

(Joseph Kolar is a free-lance writer and programmer dedicated to proselytizing for computers in general, and the CoCo specifically.)

Now for some hints from the field. John W. Bowles, Stillwater, Okla., offers the following hint.

When developing/running a program that demands maximum memory allocation, the procedure for tape users is to *POKE* 25,6:*NEW*. This clears all graphic pages reserved at power-up. However, this does not give one all of the available RAM. If the program does not require string space, a command of *CLEAR 0* following the poke will provide an additional 200 bytes of RAM. *CSAVE* the program and deliver a respectable 31,207 bytes of workable RAM in a 32K machine.

Fred Sawtelle, Huntsville, Texas, offers the following hints to owners of the CGP-115:

If your CGP-115 is in the text mode and the dip switch #3 (on the rear panel) is set to carriage return and line feed, then the control code for reverse line feed, *PRINT#2, CHR$(11)* is useless by itself. This is because the printer executes the command, backing up a line, and then automatically feeds forward a line, putting you back where you started. To prevent this, simply add a semicolon to the command, as you would to prevent automatic line feed on your TV screen:

```
PRINT#-2,CHR$(11);
```

To stop CGP-115 paper from curling up so much when it is nearly used up, flip the roll over so that the paper feed is from the top moving forward. This curls the paper backward and removes some of the tight curl.

Fred Sawtelle also offers this hint: When working out bugs in a program and listing the same lines over and over, it gets pretty boring typing *LIST* 1200—1300 (or whatever) each time. Add the following lines to your program:

```
3    GOTO 10
4    CLS:LIST 1200—1300
1302 GOTO 4
```

With this, either let the program run its course or [BREAK] and type *RUN4*.

Don Prisk, Salt Lake City, Utah, offers this tidbit:

If you want to make the *LINE* command print the color you want, type in DRAW"CN," "N" being the number of the color, before your *LINE* statement.

```
10 DRAW "C7"
20 LINE(10,40)-(30,70),PSET
```

Does your computer stuff (*i.e.*, software catalogs, invoices, newspaper clippings, *etc.*) become a disorganized pile of junk? Just use a "trapper keeper" notebook and some "trapper keeper" folders. Then, you can label each folder separately for a different software company, or whatever. Also, you can put your *TRS-80 Microcomputer News* in the binder.

Michael E. Parry, our northern neighbor from Nepean, Ontario, Canada, has some helpful hints regarding the CTR-80A cassette recorder.

To prevent wear on the keys, develop a habit of depressing the recorder keys in a certain sequence in order to avoid the ultimate breakdown. It goes like this:

Before engaging any key, hold the [STOP] key down, depress the required key. Then, while holding it down fully, release the [STOP] key.

Before operating the [STOP] key, first depress the engaged key, or keys, fully. Depress the [STOP] key and hold it down while releasing the engaged key.

It takes a bit of time to develop the habit of operating the keys in this manner. However, wear is eliminated on the internal operating surfaces of the functioning keys. Try this technique and you will immediately notice the absence of frictional resistance while operating the keys in this manner.

Michael Parry also suggests that you should never leave the [PLAY] key engaged, when tape motion is stopped. A dimple may be created on the tape. This may create a potential "drop out" type of failure. Given enough time, a bumpy capstan pinch roller may develop. Use the [PAUSE] key on the CCR-81 or the [STOP] key on the CTR-80A to disengage the capstan during lengthy pauses.

Here are a few home-grown hints:

Back to the CTR-80A! Haven't you often wished you could set the cassette counter to a desired number to avoid lots of winding and rewinding, when copying a taped program embedded far into a 60-minute tape? Save your next burnt-out cassette tape, preferably, a 60- or 90-minute one. With a black, felt-tipped pen, print "COUNTER" on both sides of the tape, after erasing or taping a new label over the old one.

Suppose you *CLOADed* a program that ended at counter number 220. You take the tape out of the cassette. Pop in a new tape and *CSAVE* it. You eject the tape and decide to put in the original tape in order to call some other program. You recall that the program ended at 220. But, the counter is at some other number.

Rewind, reset the counter and advance back to the desired location? No more!

Pop in the "COUNTER" cassette. Run in [FORWARD] or [REWIND] until the desired number appears on the counter. Stop! Eject "COUNTER." Put in the tape that you know is set at location 220. You now are properly oriented.

You don't care where you are on the counter cassette. Keep it handily located near your recorder.

The only cautionary is that you must definitely know where you are located on the tape. When in doubt, or if you are *CSAVING* with an important program, do it the hard way, until you are familiar with the system, to avoid the grief of recording over an existing program. The more you use this system, the more it grows on you.

When creating a program, it is generally good practice to assign blocks of line numbers for each routine or part of your program. Do not *RENUM* 10,10, 10 or some other equivalent, in the middle of expanding or creating a program. Even though it makes a neat, orderly numbered program, it may confuse you. You lost your assigned block series and also lost your mental line-number image of your program. This will slow down your concentration until you figure out where you are.

Consider how much easier it is to key in a program that has blocks of line numbers assigned to elements of a program. Consider the neat, but mind-numbing program that is listed numerically from 10 to 100, incremented by 1. Haven't you had the exasperating and tedious experience of keying in such a program? Wasn't it difficult to keep your place?

Since all of these hints may have been useless to you, let's have a little fun! Key in the *Evil-Eye* listing. Make a backup copy. Look over the listing! *RENUM* 1,1,1 and *LIST*. Look it over! *RENUM* 10,1,10 and *LIST* again. Look it over. Aren't you sorry that you renumbered it? Type *NEW;* get your backup copy; *CLOAD* "EVIL-EYE."

Have a little fun with *Evil-Eye*. Alter or improve it. Delete lines 220, 300, 320, 400. It may not be quicker, but you get a different presentation of the same designs.

Add the following:

```
500 FOR X=1TO3
510 CIRCLE(129,30),X,1
520 NEXT X
```

That is the scoop for putting a pupil in the top eye. Put in the other pupils.

How would you alter the routine for the four outer eyes, to make the eyes appear more round (increase the black area)? How would you make the pupils flash on and off?

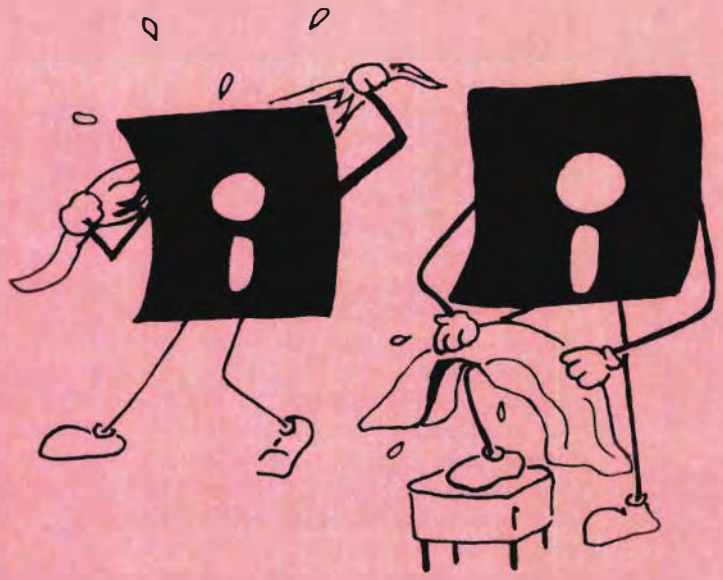
Give it your best shot and if you get frustrated or want to show me what you came up with, drop me a card at 1709 Dickinson St., Inverness, Fla., 32650. Have fun at the keyboard.

A Better Method For Washing Your Disks

By David and Thomas Szlucha

There are several ways to wash your disks. You could gently scrub them in a mild solution of diskwasher soap and water but this does not always work well because the insides of the jackets get soggy and are difficult to dry. A better way is to use the utility program presented here called *Diskwash*. *Diskwash* is a program designed to work with the Radio Shack Color Disk Operatingsystem. It was patterned after a similar program called *Wash* available to CP/M users.

Diskwash is a multifunctional disk utility designed to help maintain orderly disks and allow easy file transfer to other disks. When *Diskwash* is run, you are presented with a menu of items to select from as you step through the directory of files and programs present on that disk. You are allowed to COPY, RENAME, DELETE, or PRINT (hardcopy) the directory of the currently logged-on drive. You can also elect to change the current drive or the disk being examined. Without this utility, many of these functions can take 12 to 18 or more keystrokes to accomplish. Often, you must repeat the procedure because it is *so easy* to forget the required extension or misspell a filename. The beauty of *Diskwash* is that it is almost automatic with usually only one or two keystrokes required to carry out a function. The computer now does all of the hard work, such as remembering spelling, file extensions and the syntax of the commands. This program has not been made so automatic that you accidentally kill the wrong file; with the DELETE command you are asked "Are you sure?" So pay attention—do not respond to that cue with reckless abandonment.



The heart of *Diskwash* starts on line 140, a routine to read the names of existing files from the directory track with the very powerful *DSKI$* command. This routine was taken almost directly from Chapter 11 of the disk operating manual. Although I have read some criticism of the disk manual for lacking detail, I believe that Tandy has revealed many useful concepts which can be helpful in building handy utilities such as this. The disk manual is well worth the effort to study. There are many helpful gems amongst its pages.

Line 80 utilizes the rather well-known break key disable routine to help prevent accidental disruptions of the program. The directory printout routine in line 960 is a previously documented *POKE* which routes output automatically to the printer. It creates a single column printout exactly as it would appear on the screen.

Try washing your disks with *Diskwash*. I guarantee that once you use it, this program will become one of your most used utilities. It saves so much time. If you don't like the directory print format, change it. If you want to add a function, there is plenty of room on the screen, see lines 400—620 of the program for the input selection routine. In the way of explanation, *Diskwash* works with both single and multiple drive systems. If you are doing a COPY with a single drive system, be sure to insert the destination disk when asked, then be especially sure to remove it and reinsert the original disk when you are through. If you don't, you will only end up confusing yourself because the directory of the original disk is stored in a buffer and will be shown to you instead of the "new" disk which is in the drive. If you desire to change disks, select this option from the menu after you insert the new disk. Also, be sure to *PCLEAR1* before

---

(David Szlucha is a student in Minerva Deland High School, Fairport, New York, and is planning to pursue a career in computers. He collaborated on this article with his father, Thomas, who is a frequent contributor to ***Rainbow****.)*

loading this program, especially when you plan to transfer (COPY) large files so that the computer can use all the available memory for data storage. Because *Diskwash* is written in BASIC, if a disk error occurs, you will drop out of the program. If this happens simply retype *RUN* and try again.

Key in the program carefully and test it on a "scratch" disk to make sure that it functions correctly. You cannot be too careful with a program of this type when working with the precious information stored on your disks. Once *Diskwash* is up and running, throw away your scrub brush. Your disks will be clean and tidy and you will no longer have diskpan hands.

```
120.... 012A
250.....0262
490.... 043E
720.....0682
END... 08D7
```

The listing:

```
10 'DISKWASH 1.1
20 'DAVID SZLUCHA
30 '14 GREAT GARLAND RISE
40 'FAIRPORT N.Y. 14450
50 ' 7/11/83
60 CLEAR2000
70 B=143:S=0
80 POKE &HF8,&H32:POKE &HF9,&H62
:POKE &HFA,&H1C:POKE&HFB,&HAF:PO
KE&HFC,&H7E:POKE&HFD,&HAD:POKE&H
FE,&HA5:POKE&H19A,&H39:POKE&H19B
,&H0:POKE&H19C,&HF8:POKE&H19A,&H
7E
90 DRIVE S
100 '
110 GOSUB500
120 X=3
130 PRINT@480,"    PLEASE WAIT..
.";
140 DSKI$ S,17,X,A$,B$
150 C$=A$+LEFT$(B$,127)
160 NAM$(0)=LEFT$(C$,8)
170 EXT$(0)=MID$(C$,9,3)
180 FOR N=1 TO 7
190 NAM$(N)=MID$(C$,N*32+1,8)
200 EXT$(N)=MID$(C$,9+N*32,3)
210 TYP$(N)=MID$(C$,13+N*32,1)
220 NEXTN
230 IF FL=0 THEN N=0 ELSE N=7
240 IF LEFT$(NAM$(N),1)=CHR$(255
) THEN 310
250 IF LEFT$(NAM$(N),1)=CHR$(0)
THEN 310
260 PRINT@480,"
                ";
270 PRINT@416,NAM$(N);"/";EXT$(N
);":";
280 GOSUB380:FL=0
290 PRINT@384,"":PRINT:PRINT
300 PRINT@384,"DRIVE:"S
310 IFI$="B" THEN350
320 N=N+1:IF N<8 THEN 240
330 X=X+1:IF X<12 THEN 130
340 GOTO 120
350 IFN>0THENN=N-1:GOTO240
360 IFN=0 THEN FL=1:X=X-1:IFX<3
THEN X=11 ELSE GOTO130
370 GOTO 130
380 I$=INKEY$
390 GOSUB920
400 IF I$="R" THEN 650
410 IF I$=" " THEN RETURN
420 IF I$="X" THEN END
430 IF I$="B" THEN RETURN
440 IF I$="P" THEN 960
450 IF I$="D" THEN 730
460 IF I$="C" THEN 780
470 IF I$="S" THEN 850
480 IF I$="N" THEN 890
490 GOTO 380
500 CLS:PRINT" 'DISKWASH' BY DAV
E SZLUCHA"
510 PRINT
520 PRINT"      COMMAND      FUNCT
ION"
530 PRINT"        P          DIR TO
 PRINTER"
540 PRINT"        R          RENAME
 FILE"
550 PRINT"        C          COPY F
ILE"
560 PRINT"        D          DELETE
 FILE"
570 PRINT"        X          EXIT T
```

```
0 BASIC"
580 PRINT"    SPACE BAR     FORWAR
D"
590 PRINT"         B          BACK"
600 PRINT"         S          CHANGE
 DRIVES"
610 PRINT"         N          CHANGE
 DISK"
620 PRINT"DRIVE:"S
630 RETURN
640 GOTO 500
650 LINEINPUT"NAME ";N$:PRINT;
660 IF LEN(N$)>8 OR LEN(N$)<1 TH
EN 710
670 E$=EXT$(N)
680 PRINT@480,"         RENAMING..
.";
690 RENAME NAM$(N)+"/"+EXT$(N) T
O N$+"/"+E$
700 GOTO110
710 GOSUB500
720 RETURN
730 LINEINPUT"ARE YOU SURE ?<Y>/
<N> ";S$
740 IF S$<>"Y" THEN RETURN
750 PRINT@480,"           DELETING
...";
760 KILL NAM$(N)+"/"+EXT$(N)
770 GOTO110
780 LINEINPUT"DESTINATION DRIVE
";D$
790 IF D$>"3" OR D$<"0" THEN RET
URN
800 S1=S:D=VAL(D$):S$=STR$(S):S$
=RIGHT$(S$,1)
810 IF S1=D THEN COPY NAM$(N)+"/
"+EXT$(N)+":"+S$:GOSUB500:RETURN
820 PRINT@480,"          COPYING..
.";:COPY NAM$(N)+"/"+EXT$(N)+":"
+S$ TO NAM$(N)+"/"+EXT$(N)+":"+D
$
830 GOSUB 500
840 RETURN
850 LINEINPUT"DRIVE ";S$:S=VAL(S
$)
860 IF S>3 OR S<0 THEN S=0 :RETU
RN
870 DRIVE S
880 GOTO 100
890 PRINT:PRINT"INSERT NEW DISK
AND"
900 LINEINPUT"PRESS ENTER";Q$
910 GOTO 110
920 IF B>255 THEN B=143
930 PRINTCHR$(B);:FORP=1TO75:NEX
TP:PRINTCHR$(8);
940 B=B+16
950 RETURN
960 POKE111,254:DIR
970 RETURN
```

Can It Recite The 'Pledge Of Allegiance'?

By Steve Blyn

I can remember taking much abuse from my colleagues over the initial purchase of my Color Computer shortly after it became available. Its capabilities were basically unknown at that time. Everyone was uncertain of its abilities.

Over the past two years, all that has changed. We can each point out arcade games that graphically match those of any other computer. Alternate DOS's allow us to use very sophisticated business programs. There are also many fine educational software programs. We can certainly hold our heads high now when we compare the attributes of the CoCo with other micros on the market. But then again, there's Texas Instruments.

Why mention Texas Instruments? Not a very impressive computer at all. True, but it always talked. I have always liked their *Speak and Spell* and *Speak and Math* self-contained games. Even E.T. liked them. Early on, Texas Instruments had a *Type and Talk* program pack for their computer. I could not believe the clarity of the voice quality when I first tried it over a year ago.

Of course, it was not programmable or usable within a program. It merely repeated or parroted whatever you typed on the keyboard. But, even that was certainly impressive and something that my computer could not do. Think of the educational possibilities of a talking CoCo!

I have always felt that the addition of voice to many educational programs would be of great value and I am sure that this will be a reality in the future. Of course, voice is not necessary for all programs. A high school student studying an SAT program would no doubt be greatly annoyed and consider it a waste of time to hear the computer say, "Great, Charlie, you got this example correct." But a preschooler would love to hear that message. A balance of which programs in the future will contain voice will, I'm sure, be achieved by the various educational software companies.

There are entire groups of people that will benefit from educational voice-aided programs in the future. There is a whole class of speed-impaired people who will find voice synthesizers of great personal use. The blind, many of whom are already touch typists, will be able to more fully participate in using computer software by hearing questions and responses. Children who are too young, or otherwise unable to read will be able to participate and be rewarded in many more programs. And of course, the normal child will be attracted to using many of the new talking programs.

The situation has changed radically over the past few months. Several fine talking programs have emerged. I am referring to computer rather than cassette-generated speech. There are two methods that the companies use to create speech. One is software-based and the other is hardware-based.

The method of using software to generate speech is, of course, much cheaper: All you pay for is the program on cassette. The tape that you buy contains digitized instructions for generating phonemes or sounds. The phonemes can be combined by you into any words or sentences you desire. These programs are a real bargain. The drawback is the often criticized sound of "computer-generated voice."

(Steve Blyn teaches both exceptional and gifted children, holds two master's degrees and has won awards for the design of programs to aid the handicapped. He and his wife, Cheryl, own Computer Island.)

These programs have a distinct computer quality to them. As the author of these programs tries to improve their sound quality with more instructions and rules, he must obviously use more memory space. This leaves you little room to include your program so that the final version actually does something beyond parroting what is typed on the keyboard.

The second method utilizes a Votrax voice synthesizing chip. This is contained in a ROM pack and is plugged into the slot on the side of the computer. These packs contain a hardware voice synthesizer. A software program is included on tape to interface the CoCo to the Votrax.

I don't believe anyone would really be fooled into thinking that this type of speech were really a human being speaking, but the voice quality is truly exceptional. It is clear and understandable. The method of appending your own program to the synthesizer's program is similar in both types of voice programs. The drawback to the hardware systems has been the high cost.

The July issue of *the Rainbow* contains an article called "Cheap Talker for Our CoCo" by John R. Kelty. I was very excited after reading it. He described how anyone could purchase the Votrax chip and other parts necessary to construct their own voice synthesizer for about $80. I was all set to purchase the equipment when my wife reminded me that I am allergic to schematic diagrams. She is quite correct. After viewing them for more than 30 seconds, I begin to get dizzy. After one minute, I cry. Constructing hardware peripherals is definitely not my forte.

Recently, Bob Rosen of Spectrum Projects sent me a sample of his new Spectrum Speaker to field test. It is a hardware system containing a Votrax chip within a ROM pack and a cassette program to create the speech on a CoCo. The speech is as good as the original Texas Instruments speech that I admired. A knob in the side of the ROM pack even allows you to change the pitch of the voice at will. It has an immediate mode for voice input, a mode to save speech to cassette, and incorporating your own programs with speech are a breeze. Best of all, Spectrum is retailing the complete package tested and assembled for only $69.95.

We often find that educators lag well behind in utilizing technological improvements. Let's not let that happen this time. The technique to include your own program into any of the Voice Paks is similar. I will illustrate this from the Spectrum Speaker's program because that is the one with which I am most familiar.

The Spectrum Speaker's accompanying software contains two programs. The first is a short BASIC program whose main purpose is to access the machine language program. The ML program contains the speech-phoneme directions necessary to interface between the CoCo and the ROM pack to create speech. The ML program must be loaded before any speech can be created. This may be accomplished in either of two ways.

You may *LOAD* the BASIC program which will load the ML program for you. Then return to BASIC and type *NEW*. This will remove the BASIC program but leave the ML program intact. You may also choose to begin fresh and have your program load the ML program. This particular ML program begins at location &H2600. (This program is for the 16K version only.)

```
10 IF PEEK(&H2600)=&H7F AND PEEK(&H2601)=
   &H37 THEN 50
```

This line checks to see if the ML program is already loaded.

```
20 PCLEAR 1: CLEAR 2500,&H25FF
```

Clear memory space for the program.

```
30 CLS0: PRINT "LOADING M/L"
```

Clears screen.

```
40 CLOADM "TRNSLT16"
```

Loads the ML program.

```
50 DEFUSR2=&H2605
```

Defines the location of the text conversion program.

That's it. You are now free to create any type of BASIC program. Assign a string name to anything that you want spoken and the computer will speak it through a USR function. For example:

```
60 A$="HELLO FROM RAINBOW MAGAZINE"
70 U$=USR2(A$)
```

You may program many types of string information for your computer to speak. Here is an example of reading, printing and speaking DATA.

```
80 B$="MY FRIENDS ARE"
90 FOR T = 1 TO 5
100 READ C$
110 PRINT B$+C$
120 U$=USR2(B$+C$)
130 NEXT T
140 DATA CHERYL,STEVE,ADAM,DAVID,SHARI
```

All of the speech programs work in a similar fashion. They easily permit you to insert speech into almost any kind of BASIC program that you write. By spelling the words phonetically, you may even try any foreign language. The Votrax is above 90 percent accurate in English pronunciation and will certainly prove to be a great aid to educational programs of the future.

CORRECTIONS

Paul Hoffman reports that, "I don't know what I was thinking of at the time, but . . ." In his article, "This Peripheral Acquaintance Is A Real PAL" (September 1983, page 82), Paul gave the wrong offset address for loading a picture from tape. Instead of &HC000, you should use &HF7FF.

W. H. Barnes, of Naples, Fla., dropped in to report that a few characters are missing from the end of line 55 of the listing for Mike Dubuc's "Try These Test Patterns On Your Color Monitor." The line seems to end with GOT.

Correctly, line 55 ends with GOTO 40. It might be necessary to take out a few spaces to ensure that this last portion of a very long line stays in.

Bob Tyson, author of "Strategy Football" (August 1983, page 234), writes to say that Joe Fisher of Denver, NC, has found a "mistake."

"I call it a mistake rather than a 'bug' because it didn't crash the game or create any weird effects," says Tyson. "It just allowed for a *104-yard field goal!*"

My mistake," he continues, "was in my logic of determining the success or failure of the kick. I didn't anticipate such an attempt. The fix for it is simple (unless you like superhuman kickers)." Just change line 278 to read:

278 I=(3-2*BL)*YL*KL:IF I>=50 THEN279 ELSE 285

John Fraysse has pointed out a small glitch in his *Marathon* program which will cause big problems if not attended to. The problem is in the *DATA* statement in line 31 on page 30 of your October *Rainbow*. Remove the final datum, along with its preceding comma from line 32 — that is, the "19" at the end of the line and the comma before it. Now, change the "19" into a 198, and put it at the beginning of the *DATA* statement in line 32. Line 31 now ends with 87 and line 32 now begins: 32 DATA 198,15,(etc.).

A quirk only allows the first two digits of 198 to print at the end of line 31, so it must be dropped down to the beginning of the next line.

Firmware Hint . . .

For Your Portable . . .

If you have one of Radio Shack's new TRS-80 Model 100 portable computers, you need *PCM, the Portable Computing Magazine*. It's from Falsoft, Inc., publishers of *the Rainbow*. The new monthly is now in its fourth issue.

Poke Display

By Andrew Ilowit

Here, in the tradition of Roy G. Biv, (who is he, anyway?) is a short and simple program whose main purpose is to demonstrate a method of printing on the screen with pokes instead of print commands.

Line 10 contains the poke positions for video. Line 20 pokes each position with the ASCII code of the character to be displayed. Line 16 displays at the bottom corner of the screen the ASCII code of the displayed character. This display gets partially covered up by the last character (code 255). By using the [SHIFT] and [@] key you can ascertain the ASCII code of a displayed character.

The concept used here can become very useful for title pages for programs. In the April 1983 *Rainbow*, Jim Schmidt used these methods to display his title, and in lines 620—670 to display the binary for a Hex number.

The listing:

```
1   CLS
10  FOR X=1024 TO 1035 STEP2
15  Y=Y+1
16  PRINT@508,"";:PRINTUSING"###";Y;
20  POKEX,Y
25  FOR Z=1 TO 100:NEXTZ
27  IF Y>254 GOTO 35
30  NEXT
35  GOTO35
```

RAINBOW Info

How To Read Rainbow

Please note that all the BASIC program listings you will find in *the Rainbow* are formatted for a 32-character screen — so they will show up just as they do on your CoCo screen. One easy way to check on the accuracy of your typing is to compare what character "goes under" what. If the characters match — and your line endings come out the same — you have a pretty good way of knowing that your typing is accurate.

We also have "key boxes" to show you the *minimum* system a program needs. But, *do* read the text before you start typing.

Finally, the little cassette symbol on listings indicates that program is available through our *Rainbow On Tape* service. An order form for this service is on the insert card bound in the magazine.

The Rainbow Seal

The Rainbow Certification Seal is our way of helping you, the consumer. The purpose of the Seal is to certify to you that any product which carries the Seal has been physically seen by us and that it does, indeed, exist.

Manufacturers of products — hardware, software and firmware — are encouraged by us to submit their products to *the Rainbow* for certification. We ascertain that their products are, in actuality, what they purport to be and, upon such determination, award a Seal. This lets you know that we have seen the product and that it does, indeed, exist.

The Seal, however, is not a "guarantee of satisfaction." The certification process is different from the review process. You are encouraged to read our reviews to determine whether the product is right for your needs.

There is absolutely no relationship between advertising in *the Rainbow* and the certification process. Certification is open and available to any product pertaining to CoCo. A Seal will be awarded to *any* commercial product, regardless of whether the firm advertises or not.

We will appreciate knowing of instances of violation of Seal use.

Using Machine Language

Machine Language programs are one of the features of *the Rainbow*. There are a number of ways to "get" these programs into memory so that you can operate them.

The easiest way is by using an Editor-Assembler, a program you can purchase from a number of sources.

An editor-assembler allows you to enter mnemonics into your CoCo and then have the editor-assembler assemble them into specific instructions that are understood by the 6809 chip that controls your computer.

When you use an editor-assembler, all you have to do, essentially, is copy the relevant instructions from *the Rainbow's* listing into CoCo.

Another method of getting an assembly language listing into CoCo is called "hand assembly." As the name implies, you do the assembly by hand. This can *sometimes* cause problems when you have to set up an ORIGIN statement or an EQUATE. In short, you have to know something about assembly to hand assemble some programs.

Use the following program if you wish to hand assemble machine language listings:

```
10 CLEAR200,&H3F00:I=&H3F80
20 PRINT "ADDRESS:";HEX$(I);
30 INPUT "BYTE";B$
40 POKE I,VAL("&H"+B$)
50 I=I+1:GOTO 20
```

This program assumes you have a 16K CoCo. If you have 32K, change the &H3F00 in Line 10 to &H7F00.

What's A CoCo

CoCo is an affectionate name which was first given to the TRS-80 Color Computer by its many fans, users and owners. As such, it is almost a generic term for three computers, all of which are very much alike.

When we use the term CoCo, we refer to the TRS-80 Color Computer, the TDP System-100 Computer and the Dragon-32 Computer. It is easier than using the three "given" names throughout *the Rainbow*.

In most cases, when a specific computer is mentioned, the application is for that specific computer. However, since the TDP System-100 and TRS-80 Color are, for all purposes, the same computer in a different case, these terms are almost always interchangable.

The Rainbow Check

The small boxes which you see with programs in *the Rainbow* are our *RAINBOW CHECK* program, which is designed to help you type in programs accurately.

The check program will count the number of characters you type in. You can then compare the number the *RAINBOW CHECK* gives you to those printed in *the Rainbow*. On longer programs, some benchmark lines are given. When you reach the end of one of those lines with your typing, simply check to see if the numbers match.

To use the *RAINBOW CHECK*, type in the program, *CSAVE* it for future use, then type in the command *RUN* and press *ENTER*. Once the program has run, type *NEW* to remove it from that area into which you will be keying programs.

Now, whenever you press the down arrow, CoCo will give you the hexadecimal number of bytes in memory. This is to check against the numbers printed in *the Rainbow*. If your number is different, check the listing carefully to be sure you typed in the proper BASIC program code.

As the hexadecimal number appears in the upper-left corner of the monitor screen, you may want to clear the screen and press the spacebar five or six times to move the cursor out of the way for easy reading. The *RAINBOW CHECK* counts spaces, too, follow the spacing just as it appears in the magazine.

Here's the program:

```
 10 CLS:IF PEEK(116)=127 THEN X=32688 ELSE X=16304
 20 CLEAR 25,X-1
 30 IF PEEK(116)=127 THEN X=32688 ELSE X=16304
 40 FOR Z=X TO X+77
 50 READ Y:W=W+Y:PRINT Z,Y;W
 60 POKE Z,Y:NEXT
 70 IF W=5718 THEN 80 ELSE PRINT "DATA ERROR":STOP
 80 EXEC X:END
 90 DATA 182, 1, 106, 167, 141, 0, 68
100 DATA 134, 126, 183, 1, 106, 190
110 DATA 1, 107, 175, 141, 0, 57, 48
120 DATA 141, 0, 4, 191, 1, 107, 57
130 DATA 129, 10, 38, 44, 52, 22, 220
140 DATA 27, 147, 25, 142, 4, 0, 141
150 DATA 6, 31, 152, 141, 2, 32, 25
160 DATA 52, 2, 68, 68, 68, 68
170 DATA 141, 4, 53, 2, 132
180 DATA 15, 129, 9, 46, 4, 139, 112
190 DATA 32, 2, 139, 55, 167, 128, 57
200 DATA 53,22,126,0,0
```

RECEIVED & CERTIFIED

The following products have been recently received by *the Rainbow*, examined by our magazine staff and approved for the *Rainbow Seal of Certification*, your assurance that we have seen the product and have ascertained that it is what it purports to be.

This month the *Seal of Certification* has been issued to:

Pilot Light, a pilot light with five different colored lenses that mounts to the top cover of the computer. Comes with instructions and a lifetime warranty. Vidtron, 4418 E. Chapman, Suite 284, Orange, CA 92669, $7

Outhouse, a 32K game for one or two players. Objective: Thieves are making off with your toilet paper supply, stringing it out the door and across the field. As your ship hovers above, you must pick off the scoundrels, avoid being shot down by enemy ships, and avoid firing the fragile toilet paper. Computer Shack, 1691 Eason, Pontiac, MI 48054, disk $29.95

C-111, a utility that makes your Color Computer disks compatible with other TRS-80s and able to copy programs and data files from machine to machine without conversions or retyping. Computer Shack, 1691 Eason, Pontiac, MI 48054, disk $29.95

Balloon Attack, a Hi-Res ML game with action and adventure. The game includes a complete commented printout of the entire source code for improving programming skills. Objective: Dodge the continuing onslaught of bombs being dropped by the big red balloon flying high in the sky. Computer Shack, 1691 Eason, Pontiac, MI 48054, disk $23.95

Teee Offf, a 16K Hi-Res 100 percent ML golf game for one or two players. Objective: Select from 24 different clubs and play golf on a nine hole course while avoiding sand traps, water holes, and sanguaro cactuses. Prickly-Pear Software, 9234 E. 30th Street, Tucson, AZ 85710, tape $24.95

Disk Master, a menu-driven, several function program that: provides a speed check and adjustment to get disk drives into shape; moves data files from tape to disk—or from disk to tape—or moves any sort of file from one disk to another; prints a directory that includes ML addresses to the screen or printer; gives two different disk maps—one showing which grans are available and the other showing how many sectors of each gran are in use; and quickly kills a number of files on a crowded disk. Prickly Pear Software, 9234 E. 30th Street, Tucson, AZ 85710, disk $24.95

Mailing List, a 32K Disk BASIC mailing list program with six fields for name, address, city and state, phone number, zip code and code. It sorts alphabetically or in zip code order; searches your file on any field; adds or deletes entries, prints any or all entries, or all entries meeting search criteria; and has up to 1500 records on a single drive system. Screen display is in full upper- and lowercase in the choice of a green or white background with no adapters needed. Prickly-Pear Software, 9234 E. 30th Street, Tucson, AZ 85710, disk $49.95

Diskey, a 16K RAM utility written entirely in ML that is designed primarily to assist in recovering data from a slipped disk and for recovering KILLed files. It also can be used to examine, modify, or copy any disk, just as long as there is at least one readable sector. Includes a 50-page manual. Adventure International, Box 3435, Longwood, FL 32750, disk $49.95

Jude, a 32K ECB full text commentary and reference study program of the Epistle of St. Jude. QCS, Quality Christian Software, P.O. Box 1899, Duncan, OK 73533, tape $13.99, disk $16.99

UT-1, a 16K ECB utility for the LPVII and DMP-100 printers. It lists programs, generating 125 characters by 90 lines. **UT-1** can also be utilized with *Scripsit* by Radio Shack. QCS, Quality Christian Software, P.O. Box 1988, Duncan, OK 73533, tape $14.99, disk $17.99

Colorspeak, a user friendly, complete phoneme-based voice synthesizer that converts plain English text to speech, adds voice inflection to text to speech mode, allows programming the SCO1 directly in phonemes, spells text, and pronounces most punctuation. *Colorspeak* uses the VOTRAX SCO1 phoneme synthesizer integrated circuit as the heart of its functional electronics circuitry. This "CHIP" produces 64 different phonemes and four different inflection levels that result in 256 different sounds in which to produce speech. The SCO1 has a 6-bit data input for phoneme selection, and a 2-bit input for inflection. Includes a detailed users manual, a phoneme dictionary, and a demo tape. Bumble Bee Software, P.O. Box 25427, Chicago, IL 60625, $169

Rainbow's Corner, five 16K learning games and two math utility programs for ages five through 11. Each game gives children a chance to experiment with words or numbers in a playful environment and gives problem solving skills as children discover patterns and solutions to animated puzzles and adventures. John Boeschen & Co., 2901 Mirante, Richmond, CA 94803, tape $19.95

Atlantic Adventure, a 16/32K ECB text adventure. Objective: Try to raise your submarine safely above the water surface while you remain safely aboard. Owl's Nest Software, 9036 Pleasant Lane, Ooltewah, TN 37363, tape $21.95

Bomber Command, a 16K 1941-45 strategy war game. Objective: Control the allied air armada and destroy Germany's industry by launching bombing raids each turn, which equals one real month during the war. Play is held on a tactical map that may be toggled to a strategic map via a ML routine and semi graphics. Ark Royal Games, P.O. Box 14806, Jacksonville, FL 32238, tape $19.95

Microdis, a ML program designed specifically for the 4K MC-10 computer, that will allow you to read and disassemble areas of RAM or ROM in the MC-10, and understand the complete MOTOROLA 6803 OPCODE set and will disassemble areas of memory to your printer or screen, using industry accepted mnemonics. Micro Ten Software, 496 Amboy Avenue, Perth Amboy, NJ 09861, tape $19.95

Pre Reader, a 32K ECB learning tool for preschool (ages 3-5) children and beginning readers (grade 1). Level I teaches the child to work with colors, shapes, numbers, and upper and lower case letters. It also teaches how to associate sounds with the letters which represent them. Level II matches single letters and consonant blends with their corresponding sounds. Sugar Software, 2153 Leah Lane, Reynoldsburg, OH 43068, tape $19.95

Data Doctor, a 32K menu driven program that provides the file edit capabilities most DOS provide. It lists files, examines, edits and deletes data. It also corrects file errors without running an entire system. Superior Graphic Software Products, P.O. Box 451, Canton, NC 28716, disk $49.95

The 8010 Interface, a serial interface for the Gemini-10 that will link virtually any personal computer with an RS-232 serial port, to a Gemini-10/15. Dip switches select Baud rate, parity, word length and start bits. The 8010 is shipped configured for TRS-80C, 4800 Baud. Includes an assembled and tested circuit board with mounting hardware, a five foot data cable for the TRS-80C and instructions. Dayton Associates of Willie Hall, Inc., 7201 Claircrest Drive, Building D, Dayton, OH 45424 $27.50

Remrem, an educational game with eight skill levels. Objective: Try to remember a sequence of colors that the computer has selected with sounds. Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada L4G 2H6, tape $20.

Battle, a war game. Objective: Score points by bombing and sinking boats while avoiding hitting bombs. Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada L4G 2H6 tape $20.

Romdisk, a program that will allow you to load Radio Shack ROM Packs from a disk. Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada L4G 2H6, tape $20.

Mr. Copy, a copier written in ML that will make backup tape copies, and is capable of making up to 99 copies in one loading. Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada, L4G 2H6 tape $25

Concen, an educational game in four skill levels. Objective: The computer hides eight pairs of three letter words that you must try and match up against the computer or an opponent. Similar to the television game of "Concentration." Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada L4G 2H6 tape $20

Colormind, an educational game for up to four players. Objective: The computer hides a sequence of four colors. You must guess the colors and the position they are in. Similar to a popular board game. Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada L4G 2H6, tape $20

Cribbage, a 32K hi-res graphics card game for two to four players. Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada L4G 2H6, tape $20

Stockbroker, a 16/32K ECB game for up to six players. Objective: Each player is given $500,000 to buy stock and must play and keep their stocks, as they may go up or down during the game. Aurora Software, 49 Brookland Avenue, Aurora, Ontario, Canada L4G 2H6, tape $20

Mastermail 1200, a 32K disk mailing list program for serious business applications. Can store up to 1200 five-line labels per disk, print by code, and uses a ML sort. Spectrum Projects, 93-15 86th Drive, Woodhaven, NY 11421, disk $49.95

Microterm, a terminal program for the new Radio Shack MC10, used to access bulletin boards, CompuServe and other data lines. Spectrum Projects, 93-15 86th Drive, Woodhaven, NY 11421 tape $24.95

Kron, a 32K ML game with four playing screens. Objective: Select from screen of blocks, spiders, cycles or tanks and achieve the best possible score. Similar to the arcade game *TRON*. Oregon Color Computer Systems, P.O. Box 11468, Eugene, OR 97440, tape $26.95

Wild Party, a 16K adult party game for two to six couples. The play of the game consists of a sequence of events performed by the players designed to liven up the dullest party. B&B Software, P.O. Box 210, Jenkintown, PA 19046, tape $35

Family Finance, four 4K BASIC programs for the MC-10, including *Mortgage, Amortization, Loans, Phone Numbers* and *Checkbook Balancer*. Family Computers, 4125 Prescott Street, Sarasota, FL 33582, tape $14.95

Family Fun, four 4K programs for the MC-10 including *Forest Fire, Simple Simon, Concentration and Master Mind*. Family Computers, 4125 Prescott Street, Sarasota, FL 33582, tape $14.95

SRC, Screen Reference Card, a 16K aid to BASIC programming. It offers a variety of information normally available in manuals or reference cards such as: a quick reference to the proper syntax of all commands; shows text screen (PRINT@) locations; explains the coordinate grid patterns; displays graphic character codes; displays ASCII character codes; is user defined; etc. CoCo Data, 1316 Quail Avenue, McAllen, TX 78504, tape $7.95

CSP-F, Command Stream Processor — Full, a utility which will allow the user to pre-package a series of input lines to be automatically passed to BASIC. These lines may be direct commands or values requested by INPUT or LINE INPUT statements of a BASIC program in execution. The end result is to allow the automation of a total stream of activity and may be considered a program to run your other programs. The user is supplied both a short processor version, for use when RAM space is tight, and a full version which allows a more sophisticated approach to command stream programming. Custom Software Engineering, Inc., 807 Minutemen Causeway, CoCo Beach FL 32931, tape $19.95

Gomoku/Renju, 16K games of strategy for two players in eight skill levels, played on a pattern or grill of intersecting lines. Objective: Two players move alternately by placing a piece of their own color on any vacant point where two lines cross, and be first to create a horizontal, vertical, or diagonal line of five adjacent pieces of his own color. Radio Shack stores Nationwide. Rompack $19.95

Reactoid, a 16K game of skill. Objective: You are in manual control of the reflectoid in a computer guidance system and must keep energy particles from hitting other particle emission tubes that will eventually melt the core of the reactor and must prevent self destruction and meltdown. Radio Shack stores nationwide, Rom Pack $19.95

Dark Castle, a 32K ECB text adventure program. Objective: You must free King Lothan and return him to his rightful throne. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92616, tape $14.95

Witches Knight, a 32K ECB text adventure program. Objective: you must free Sir Noble from the witches evil spell while avoiding many obstacles during this quest. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $14.95

Scavenge Hunt, a 32K ECB text adventure. Objective: You must return all the items on the scavenge hunt list to Hickory Ridge, in order to free your niece, Rebecca from the evil hermit of Medicine Tree. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $15.95

Stalag, a 32K ECB text adventure program. Objective: You must get out of a prison camp in Germany, 1944, alive. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $14.95

Evasion, a 32K ECB text adventure program. Objective: You must evade the German patrols and snipers. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $15.95

Bomb Scare, a 32K ECB text adventure program. Objective: You must disarm all eight bombs that have been planted in the city by a terrorist group. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $14.95

Beacon, a 32K ECB text adventure program. Objective: you must successfully operate an old lighthouse in order to avert any major sea-going accidents. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $14.95

Mansion of Doom, a 32K ECB text adventure program. Objective: you must successfully reunite the Crown Princess Marlena with the townspeople of her village in Transylvania. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $14.95

Sac, a 32K flight simulation program in nine skill levels. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $19.95

Isle of Fortune, a 32K ECB text adventure. Objective: You must find a treasure and bring it back to the waterfront bar where the adventure starts. Pal Creations, 10456 Amantha Avenue, San Diego, CA 92126, tape $19.95

Option 4, a 32K ECB Canadian payroll program which features all provinces and territories; auto S.I.N. validation; formatted screens, full format trapping; all media transfers; detailed pay stubs; batched checks; total cost/duty allocations; and gross pay any period. YGS, Box 208, Brechin, Ontario, Canada LOK 1BO, disk $149.95

The *Seal of Certification* program is open to all manufacturers of products for the TRS-80 Color Computer, the TDP-100, or the Dragon-32, regardless of whether they advertise in *the Rainbow*. By awarding a *Seal*, the magazine certifies the program *does exist*, but this *does not* constitute any guarantee of satisfaction. As soon as possible, these hardware or software items will be forwarded to *the Rainbow's* reviewers for evaluation.

—Jutta Kapfhammer

Grab your modem, tell the kids to use the neighbor's phone, and get yourself ready to transcend the solitary confines of your CoCo cloister, 'cause here come

The Bulletin Boards

	Name of BBS	SYSOP	Phone	Comments
ALASKA				
Fort Wainwright	CoCo-Cold BBS	Kerry Clabaugh	907-ELO-COCO	24 hours
ARIZONA				
Phoenix	6809 CCUG	———	602-298-4194	System down until Nov. 1
CALIFORNIA				
Berkeley	East Bay BBS	———	415-895-0699	
Burbank	The Fantasy Plaza	———	213-244-1100	Shop for hardware, software, health, beauty aids & kit
Los Angeles	———	Norm Wolfe	213-502-0907	
Los Angeles	Magnetic Fantasies Board	———	213-388-5198	Science fiction films, literature software
		Don Brown	213-563-7727	
Napa	Morning Star Color BBS	Phil Rusin	707-257-1485	
San Diego	JARB/COCOSig BBS	Joe Bennett	619-474-8981	
Sunnyvale	Silicon Rainbow BBS	Shawn Jipp	408-733-6809	
CONNECTICUT				
Greenwich	Education-80	———	203-629-4375	Education conferences in northeast & new developments for computers in education
Meriden	Cool CoCo	———	203-237-2668	
FLORIDA				
Ft. Lauderdale	Color Burst		305-525-1192	
Lantana	———	Bob Boyce	305-533-0333	
Orlando	All Systems Go		305-894-1886	
Pensacola	Dr. D's CoCo Corner	Gary Dunsford	904-456-7195	24 hours a day/programs for creating graphic displays, disk utility, games, etc.
Sarasota	Color-80 BBS (Silicon Rainbow Products)	Ernie Bontrager	813-924-COCO	24 hours a day
West Palm Beach	The Notebook	———	305-686-4862	An exchange of messages related to freelance assignments between writers/editors

GEORGIA				
Atlanta	Bulletin '68	Wayne Ashe	404-929-0680	
Atlanta	CoCo Board	Lee F. Blitch	404-378-4410	Between 6 a.m. and 6 p.m.
Augusta	Forum 80	———	803-279-5392	
Ringgold	———	Dennis Womack	404-891-0136	
ILLINOIS				
Chicago	OS/9 Board	George Doner	312-397-8308	
Wheaton	Metro West Data Base	Skyline Marketing	312-588-7917	
IOWA				
Iowa City	Apple-Med Bd.	———	319-353-6528	Medical conferences plus related hardware and software
KANSAS				
Kansas City	Hot CoCo BBS	John Ross	913-384-2196	
Kansas City	Dickinson's Movie Guide	———	913-432-5544	Movies playing in area and a synopsis
		Steve Odneal	816-358-6222	
KENTUCKY				
Louisville	Baud-Ville	Computer Emporium	502-423-0695	
LOUISIANA				
New Orleans	N.O. BBS	Mike Randazzo	504-277-9450	
MASSACHUSETTS				
Arlington	Color-80 BBS (Silicon Rainbow Products)	Greg Moore	617-646-6809	24 hours a day
Woburn	———	Orman Beckles	617-321-6809	
MARYLAND				
Greenbelt	Gas-Net Board	———	301-344-9156	Keeps up on space information
Marlboro	Color-80 #25	———	301-599-1726	
MICHIGAN				
Dearborn	DSL BBS	Dennis S. Lewandowski	313-582-3406	24 hours a day, free board operated by DSL
Detroit	Westside Down	———	313-533-0254	
Grand Rapids	Tom Mix Software	Tom Mix	616-364-8217	24 hours a day
Lansing	———	Greg Miller	517-339-3367	
MINNESOTA				
Minneapolis	NC Software	Bob Shaw	612-533-1957	
Moorhead	CoCo BBS	———	701-280-1928	24 hours a day
MISSOURI				
Independence	MACCUG	Jerry Morgan	816-358-MACC	24 hours a day/seven days a week
Kansas City	Mid America CC	Steve Odneal	816-358-6222	
NORTH CAROLINA				
Greenville	Sangarnet Bulletin Bd.	Gary L. David	919-758-5261	
NEW JERSEY				
Atlantic City	Joke Byte Board	———	609-927-5922	Some bad jokes
New Brunswick	CoCo Board	———	201-572-0617	
NEW YORK				
Glen Oaks	CoCo's Next	———	212-423-4623	
New York	Nybbles-80	———	212-626-0375	
Rochester	———	———	716-899-4473	
	Color-80 BBS	———	716-381-6800	
Woodhaven	Rainbow Connection #1	Bob Rosen	212-441-3755	
Woodhaven	Rainbow Connection #2	Bob Rosen	212-441-3766	
Woodhaven	Rainbow Connection #3	Bob Rosen	212-441-5719	
OHIO				
Cincinnati	Cintug TBBS	Dan Jaffee	513-522-8227	
		Jeff Butler	513-742-1895	
Youngstown	CoCo Club 80	Larry Cadman	216-788-7910	

OKLAHOMA				
Oklahoma City Oklahoma City	Flexnet (Flex Sys.)	Roger Walton/ R. L. Hilbun	405-722-6809	
OREGON				
Portland	Bit Bucket Systems	Rick Bensene	503-761-6345	
PENNSYLVANIA				
Erie	CoCo BBS	———	814-898-2952	
Philadelphia	CoCo BBS	———	215-857-3035	
RHODE ISLAND				
Providence	Syslink-80	Andy Nulman	401-272-1138	
TENNESSEE				
Hixson	68XX Micro Journal BD	Don Williams & Tom Williams	615-842-6809	
TEXAS				
Elgin	Color Connection	Peter Banz	512-285-5028	
Henderson	Int'l C.C.C.	Ron Garrett	214-657-8147	
Houston	CoCo Country	———	713-331-2599	
Houston	Mines of Moria	———	713-871-8577	
WISCONSIN				
Milwaukee	The Big Top Games	———	414-259-9475	
WEST VIRGINIA				
Morgantown	Mountaineer Softline	Wallace Coyler	216-788-7910	24 hours a day/seven days a week
CANADA				
ONTARIO				
Burlington	———	Brent Bogle	416-639-3812	

Thoughts of a Computer Widow

Or What's A Nice Girl Like Me Doing With An I/O Error?

By Charlotte Bish

It has been one of those days here! My rocking chair broke, I burned three fingers and some clown has written "I love my CoCo" in the dust on the table. I may not need to tell you this, but, it was the clown I married.

This gave me reason to pause and reflect on our past nine years. There have been many tests of our marriage; my inability to cook and two new babies were only the tip of the iceberg!

I didn't mind when he had radio control planes buzzing around day and night . . . I didn't mind when he left me home alone to go and see *Star Trek* . . . I was even nice when he left me to take a computer course—on the night of our eighth anniversary, no less!

But soon there was to be a new challenge to our love! He bounded in one evening, "hi baby!" (Why do I always know something is up when he says that?) He continued, "I know exactly what we can do with our income tax refund money!" I stood in a somewhat fearful silence, and he concluded, "let's buy a computer!" He did—sure enough.

One morning the United Parcel man knocked on our door with a big red and white box. I wonder if I should mention this, but I can't keep it quiet. On that very same day I was getting the Asian flu—I should have taken that as an omen.

Nonetheless, we are now a one-computer family. You will hardly be able to imagine my life since then. But try! Please try!

The only way I know him now is from the back of his head. I used to lose him if he got one aisle away from me in the supermarket—no more. Now when I lose him in town, the first place I go is Radio Shack, then the newsstand. If I haven't found him by this time, I simply stand on any corner, face any direction and yell, "Willis, if you aren't here in 30 seconds I'm going to sit on your computer!" I bet O. J. Simpson's mother wished O. J. could run half that fast.

His first question used to be, "what is there to eat?" This has been replaced by, "did my *Rainbow* come yet?"

I have gotten used to carrying his computer (strike that—"our" computer) almost everywhere we go. This is inconvenient, but it cuts down on his acute computer withdrawal. We take CoCo to his mom's house, to my mom's house, virtually everywhere but the grocery store. He'd take it there too, but there is no place to plug it in. I was a little worried about his "hobby" causing family squabbles but everyone was quite receptive. As my mother put it, "Charlotte, be thankful he doesn't drink!"

As I come to a close, I was concerned that you might think that I dislike my love's new passion—well, "byte" your tongue! Besides, how can anyone not like something called "CoCo?" After all, what is so bad about a computer that does all the things every computer can do, and (an added bonus) keeps my dearest busy, happy, contented, and out of my hair for hours? It even teaches my kids math, numbers, letters, remembers my recipes (I can't remember my recipes) and on top of all that it plays me music, times my baking, and draws me pictures. What do you know—all this and heaven too!

CoCo Clubs

new clubs

Editor:

I would like to announce the Spartanburg County CoCo Club formed in March 1982 with an excess of 35 members at present. We meet every Thursday evening at 7:30 p.m., in the old library, wing C of Spartanburg Technical College, Spartanburg, S.C.

For more information, call or write Dennis Shattuck, 473 Royal Oak Dr., 29302, or phone 803-583-3017.

Dennis Shattuck
Spartanburg, SC

Editor:

Deer Park Color Computer Club is for CoCo users in the Deer Park, Pasadena area of Texas. Our club meets on the third Sunday of every month. We welcome anyone who is interested to contact Don Burr at (713) 479-6657 or (713) 479-5313 or write me at 4314 W. Grant , 77536. *Rainbow* is by far the best magazine written for the CoCo!

Don Burr
Deer Park, TX

Editor:

I would like to inform your readers of a new Color Computer Club for the users in the Radcliff, Ky. area. We meet on the second sunday of each month 2:00 p.m. at the Kentucky Utilities Bldg. For further information please contact: The Radcliff Color Computer Club c/o N4GSB Bryan Harp, 287 Highland Dr., 40160.

Bryan Harp
Radcliff, KY

Editor:

I have founded a Color Computer Club in the desert area. It is called the CoCo Nutz Computer Club. We meet every two weeks at Allstate Savings In Indio. This group is also open to all other types of computer owners. We are about to publish a newsletter that will contain review of hardware and software for different systems.

Walter V. Seay
Thermal CA

Editor:

This is an announcement of the existence of the Midlands 80 Computer Club of Columbia, S.C. We currently have about 100 members (60 CoCo users) and are extremely active. Anyone interested should write to: President, Midlands 80 Computer Club, P.O. Box 7594, Columbia, SC 29202.

Jerry Kilpatrick
Columbia, SC

Editor:

This is to announce the formation of the Shippensburg Color Computer Club "Rainbow Connection." We wish to invite the people of the Southcentral Pennsylvania area to join us at our regular monthly meetings which are held on the fourth Tuesday of each month in Room 104 of the Franklin Science Center, Shippensburg University.

For further information, please contact: Shippensburg Color Computer Club, c/o Department of Mathematics & Computer Science, Shippensburg University, 17527, (717) 532-1406.

Veronica H. Mowery
Secretary-Treasurer
Shippensburg, PA

Editor:

I would like to announce the formation of the Ventura County Color Computer Club, sponsored by the Oxnard public library. The club meets the third Wednesday of each month — place to be annouced.

For information, contact me at the library, 805-984-1842. We are sharing a bulletin board service with the Cabrillo Computer Club, so anyone with a modem can leave a message for Pete at 805/987-2577.

Doug McLaughlin
Oxnard, CA

Editor:

A Color Computer group has begun locally which serves southern Michigan and northern Indiana . . . the name of the group is "Michiana Color Computer Club" and further information can be obtained from Clay Howe, Secretary.

Clay Howe
Sturgis, MI

Editor:

I am pleased to let you know that Central Arkansas now has a Color Computer Club. The CACCC meets twice monthly, is informal and free. More information can be obtained by writing or calling Melinda Braslavsky, 1205 Erving Rdg. Tp., 72023, 982-8854. Many thanks.

Melinda Braslavsky
Cabot, AR

Editor:

I would like to announce the formation of the CO*CO*M*U*G* of Marion, Ind. (Color*Computer*Marion*Users* Group). We are meeting on the second Monday of each month at 7:00 p.m. at the Marion public library. The CO*CO*M*U*G of Marion, Ind. will serve the interests of all present and prospective Co*Co owners in Marion and Grant County.

Anyone wanting more information about this growing users' group may contact me at 3635 N.300 East, 46952 or just show up at the meeting. I would also be interested in hearing from other clubs in Indiana. Your truly,

John A. Helwig
Marion, IN

Editor:

I am pleased to announce the formation of an Adventure Game users club. We are hoping to involve people from across the continent who are interested in adventure gaming. We will try to establish a library of member-written games as well as offering hints to the solutions of the many games now on the market and to offer reviews of new games. Anyone interested should contact me at the following address: 35 Scotland Road, Agincourt, Ontario M1S-1L5.

Ian Hanna
Canada

Editor:

Please include our club name and mailing address in your magazine. BUG-80 Users Group, P.O. Box 62, 08826.

George Miller, Secretary
Glen Gardner, NJ

Editor:

I would like to tell about the forming of a Color Computer Club in the Adirondack region. The club would like to hear from anyone — anywhere who would like to join. Please write to me at Box 365, 12814, or call me at (518) 644-9927. Thank you.

Bill Edwards
Bolton Landing, NY

Editor:

Well, I guess I have waited long enough for a Color Computer Club to be started in my area. Instead of waiting another year I have decided to start one myself. Anyone in the northern San Diego County area is welcome to contact me at 743 Santa Paula, 92075 or call (691) 755-2961.

Don Bradford
Solana Beach, CA

Editor:

The Mid-America Color Computer User's Group (MACCUG) c/o Bob Colin, 716 Crisp, Independence, MO 64054, phone (816) 833-0367. We have been in existence for a little over a year and have approximately 90 members throughout the K. C. Metro area. Our meetings are held every 4th Tuesday at 7 p.m. in the Campus Center Building (room 503) at Penn Valley Community College, 3201 Southwest Trafficway, Kansas City, MO.

Jerry Morgan
Independence, MO

Editor:

I would like to inform all your readers that Saskatoon (Saskatchewan, Canada) has a Color Computer Club. The SCCC (Saskatoon Color Computer Club) meets on the second and fourth Saturdays of each month at 2 p.m. at the J.S. Wood Branch Library downstairs. Anyone interested can write to Guy Tomashewski, 415-423 Pendygrasse Rd., S7M 4Z2. Yours truly.

Guy Tomashewski
Vice President SCCC
Saskatoon, Saskatchewan,
Canada

Editor:

The Keesler Color Computer Club has been formed and is holding bimonthly meetings for CoCo owners in the Biloxi Mississippi area. Meeting times are the first and third Monday of each month. For more information call Tony at (601) 374-3375.

Tony Byorick
Biloxi, MS

Editor:

The East Bay Color Computer Club has only met once and is situated in the East San Francisco Bay Area. The meetings are probably going to be in Oakland. It is not an official organization and so, to find out about it, you would have to log on to a BBS and ask around.

Justin Paola
Berkeley, CA

Editor:

Raleigh, N.C., has a Color Computer Club with over eighty members, forty of which always show up at each meeting.

We meet at 7:30 p.m. on the second and fourth Wednesdays at a local school. We have a club library of over forty uncopyrighted, copyable programs that have been written by the club members for the club members' use. There are no dues yet, but the thought and need for a newsletter may prompt us to start a small fee in the future.

Newcomers are always welcome. We love to answer the questions that new owners often have: P.O. Box 681, 27529.

David Roper
Garner, NC

Editor:

I am writing in the hopes of finding other readers who are interested in the New Radio Shack MC-10 MicroColor Computer. I have just purchased this new machine, and would like to start a user's group. Write me at 36 Prospect Avenue, 10562.

Bob Kanto
Ossining, NY

Editor:

Several CoCo owners and I are forming a club called "Color Swap" which deals in trading color computer machine language programs. Anyone interested can reach me at the following address: 12541 Norman Rd., 48097.

Timothy Wehner
Yale, MI

Editor:

I would like to announce the formation of a Color Computer Club for the New York City Metropolitan area. We would gladly accept people from New Jersey and Long Island. If you are interested in joining the club, please contact me at (201) 889-5737 anytime during the week.

Bob Shichman
Scotch Plains, NJ

Editor:

I am interested in forming a CoCo user's group in the Oakland area. If you are too, or are interested in membership, please contact me at (415) 487-3537.

Joe Hayden
Hayward, CA

Editor:

I am very much interested in forming a C.C. User's Group in the Medina, Maple Plain, Mound Minnesota Area. Write me at 3145 Cedar Ave., 55359, or call (612) 479-2746.

Jim Beletti
Maple Plain, MN

Editor:

A couple of guys, in cooperation with KCR Data Co., are interested in starting a local Color Computer user's group/club in and around Glendale Heights, Ill. Anyone interested can write to me c/o KCR Data Co., P.O. Box 1140, 60139 or call 668-0629.

Keith R. Serue
Glendale Heights, IL

Editor:

We're trying to start a CoCo Club in the Warsaw, Ind., area. We're real *Rainbow* fans and hope some of your readers in Kosciusko County will join us. Contact me at RR #1, Maze Road, 46580.

Steve Hardin
Warsaw, IN

Editor:

I am interested in forming a nationwide *kids* CoCo Club. For kids and kids only. We could exchange programs and talk over ideas (through the mail). For more information write me at 914 Albany Ct., 27609.

David Joyner
Raleigh, NC

Editor:

I am interested in starting a Color Computer Club for kids in the Amherst/Buffalo area. If anyone else is interested, please contact me at 88 Ruskin Rd., 14226, or call (716) 836-0713.

Devon Copley
Amherst, NY

Editor:

I would like to know if there are any local Color Computer Clubs near here. If anyone has any information, please contact me: 477 Rusty Lane, 83301.

Jim Reynolds
Twin Falls, ID

Editor:

I would like to start a San Diego CoCo-Nut club for people like me who live too far away to joing the club at Jarb Software.

If you are interested in joining, call me anytime before 6:00 p.m. at (619) 483-7349. Or write to 727 Sapphire St., #105, 92109.

Matthew Murray
San Diego, CA

Editor:

I am looking for a CoCo club in southwest suburban Chicago. If anyone has any information, or is interested in starting one, call me at (312) 852-5432.

Jim Barbour
Downers Grove, IL

Editor:

We would like to start a Color Computer club in the Rutland, Vt. area., to share programs and help each other to become more computer capable.

Anyone interested in joining can call me (802) 885-4807 or Karl Towsley at 446-2184.

Joan Cacioppi

Editor:

I am interested in joining a Color Computer Club on Long Island or in Queens. Write me at 35 Dorothy Lane, 11754.

Allen Smith
Kings Park, NY

Editor:

I would like to form a CoCo club for people living anywhere in the U.S.A. The main idea is to create and exchange programs. For more information contact me at 12 Cardinal Place, 19610.

Matt Duffy
Wyomissing, PA

Editor:

I would like to announce the formation of the Blennerhassett Color Computer Club for the Parkersburg, West Virginia/Marietta, Ohio area. Everyone welcome. Large program library, excellent technical assistance available. For more information please write to me, David Greathouse, President, Route 9, Box 119, 26101.

David Greathouse
Parkersburg, WV

Editor:

Now a Color Computer User's Group in the eastern Pennsylvania area. We have members from the Allentown, Bethlehem, Easton, Pa., and Phillipsburg, N.J. areas. For more information call Bill Jones at (215) 253-5733 or me at 434-6387.

Jerry Behler
Allentown, PA

Editor:

Broome County (New York State) now has an organized and fully-functioning users group for Color Computer owners: Broome CoCo Club, c/o Bucky Helmer, President, 57 Front Street, 13905, 723-8223 or 724-5726.

The Broome CoCo Club meets the first Thursday of each month (yes, even through the summer—you know how computerists are!) at 7:15 p.m. in the NYSE&G Corporation Service Center, Old Vestal Road. Membership is open to individuals or families at $12 per year.

Bucky Helmer
Binghamton, NY

Editor:

I need someone from a Dallas area CoCo club to contact me at (214) 522-5548.

Michael Gwynn
Dallas, TX

Editor:

Finally! A CoCo User's group has formed in the Colorado Springs area and everything really looks promising. Interested parties please contact me at (303) 597-7806.

Herbert B. Ridge
Colorado Springs, CO

Editor:

I would like to meet anyone in the Tri-Cities Area who is interested in a Color Computer Club please call me at 586-4840.

Thell Rooney
Keenewick, WA

Editor:

We would like to announce the formation of the Northwest Florida CoCo Nuts (Color Computer Users Group). We meet on first and third Fridays of every month at 7:30 p.m. in Fort Walton Beach. For more information contact Bill Lamb (904) 244-5281 or me at (904) 837-6538.

Jim Waits
Destin, Florida

Editor:

I'd also like to inform all CoCo users of the Mississauga area of the Official 80C club. Any interested in joining may write to 2422 Old Carriage Rd., postal code L5C364.

Vincent Lok and Roland Hendel
Mississauga, Ontario

Editor:

We would like to inform your Fort Wayne, Ind. readers of the formation of the Three Rivers Users Group. Our meetings are held on the third Thursday of each month in the Colony Bay Apartments meeting room. The informal meeting is from 5:00 to 7:00 p.m. with the formal meeting starting at 7:00 p.m. Hope to see some new CoCoNuts at the next meeting!

Three Rivers Users Group
Fort Wayne, IN

Editor:

We are proud to announce the formation of a new Color Computer Users Club for San Bernardino/ Riverside and vicinity. The Citrus Color Computer Club (CCCC or 4Cs) invite those with TRS-80 CoCo, TDP-100, and Dragon to join our membership. Individual member ship fees are $12 for one year and family membership fees are $20 for one year. For more information, please contact us by writing c/o Personal Relations Chairman, 18227 Muriel Avenue, 92407.

Michael J. Schindler
San Bernardino, CA

Editor:

I am pleased to announce the formation of the Topeka Computer Club. We have over 25 members and would like to have more. If you are interested wrtie me at 2224 Hope, 66614, or call 272-1353.

Kevin Cronister
Topeka, KS

Editor:

This is to inform your readers of the existence of the "LOCO-GOCO," a new users group for the CoCo, in Louisville. For more information call 1-502-458-6690 or 1-502-458-0649.

Mike Standefer
Louisville, KY

Editor:

Would you please announce our new address for the Halifax-Dartmouth Color Computer Users' Group in your next issue? The group has been meeting on the third Monday of every month since February in the Dartmouth Regional Library Auditorium at the rear of the building. Meetings are from 7:00 to 11:00 p.m. and our club project is a bulletin board. Our new address is Roger Pocklington (902-469-3656), President, P.O. Box 572, B2Y 3Y9. The group is thriving with over 20 families having attended meetings so far.

Bob Hamilton
Dartmouth, Nova Scotia

Editor:

Forming Color Computer Users Group in the Simi, Westlake, Agoura, Woodland Hills, Thousand Oaks, Calif. area. If interested, please contact me at (805) 497-7268.

Bruce Rothermel
Westlake Village, CA

Editor:

We are the Metro Area Color Computer Club — serving Omaha, Nebraska/ Council Bluffs, Iowa. Contact Kurt Knudtzon, 3324 11th Avenue — Council Bluffs, IA 51501, (712) 323-2084. Thanks.

"Master of Ceremonies"

Editor:

I would like to join to a CoCo Club in the U.S.A. Everybody interested should contact me at Rue Alagoos, 16/408, CEP 30.000

Francisco Joanes
Belo Horizonte MG, Brazil

Editor:

I am planning to start my own computer club. The name would be The 4C Club: The Conglomeration of Color Computer Computerists' Club. I have a friend in my area that owns a CoCo, and we plan to find more members. Write me at 472 Sundance Street, 91360.

Robert Rimmer
Thousand Oaks, CA

Editor:

Since I am unable to locate any CoCo Clubs in the Baltimore, Md. area, I am considering starting one myself. If you know of anyone else around the Baltimore area who would be interested in starting a club please contact me at 9H Quiet Stream Court, 21093 or call (301) 628-0445.

Marc D. Behr
Timonium, MD

Editor:

I would like to start a Color Computer Club in and around Baton Rouge, La. Anyone interested, write me at 8929 Metairie Dr., 70810 or call (504) 293-7799.

Gary Cash
Baton Rouge, LA

Editor:

I'd like to inform your readers that since April of this year, the Los Angeles-Wilshire Color Computer Users Group, known by members and others as "LAW," has been meeting on the second Saturday of every month. Interested users can leave a message on our bulletin board which is on-line 24 hours a day, seven days a week. The modem number is 502-0907. The owner and SYSOP, Richard Lack, wants you to feel free to explore the system.

Please bear in mind that we are a users group and not a club, so informality is stressed! Anyone wanting further information can call or write to us at P.O. Box 11151, 90213 or call (213-389-3344).

Norm Wolfe
Beverly Hills, CA

Editor:

We would like to announce the formation of the Northwest Florida CoCo Nuts (Color Computer Users Group). We meet on first and third Fridays of every month at 7:30 p.m. in Fort Walton Beach. For more information contact Bill Lamb (904) 244-5281 or me at (904) 837-6538.

Jim Waits
Destin, Florida

Editor:

I'm looking for a group of people in the Edmonton, Alberta, Canada area who I can do CoCo talk with and exhange programs and ideas. Write to 5204-90 Avenue, T6B ON9.

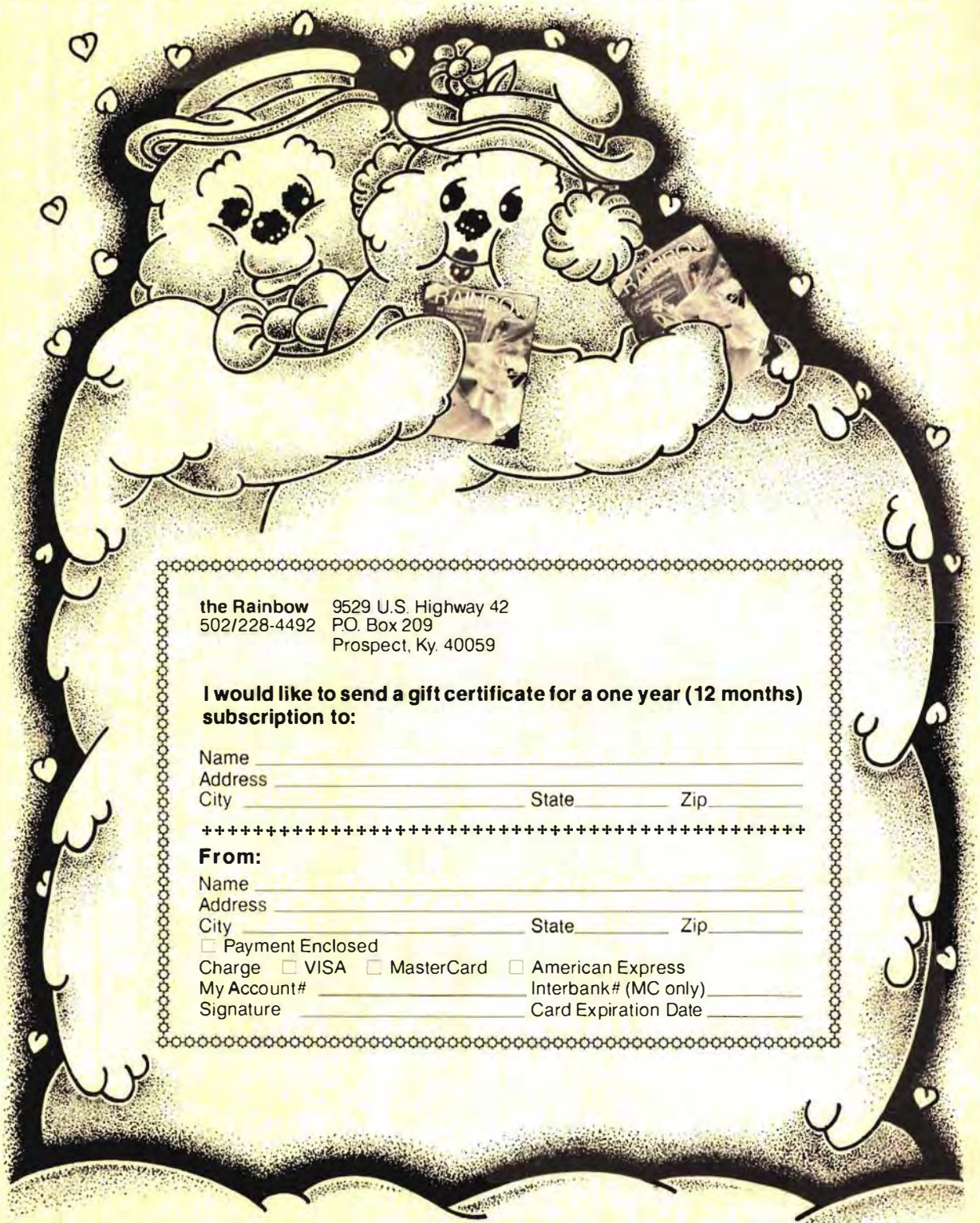
John Gaudin
Edmonton
Alberta, Canada

Editor:

We have just formed a CoCo users' group in the Dayton, Ohio area. Dayton Color Computer Users' Group, 609 Applehill Dr., West Carrollton, OH 45449, PS# 513-859-3529.

Joseph P. Evans
West Carrolton, OH

It'sno job to give the perfect gift for this Christmas.



the Rainbow 9529 U.S. Highway 42
502/228-4492 P.O. Box 209
Prospect, Ky. 40059

I would like to send a gift certificate for a one year (12 months) subscription to:

Name _____
Address _____
City _____ State_____ Zip_____

++

From:

Name _____
Address _____
City _____ State_____ Zip_____
☐ Payment Enclosed
Charge ☐ VISA ☐ MasterCard ☐ American Express
My Account# _____ Interbank# (MC only)_____
Signature _____ Card Expiration Date _____

Get the ball rolling and fill out the form above to give someone a RAINBOW certificate in time for the holidays.

Subscriptions to **the RAINBOW** are $22 a year in the United States. Canadian and Mexican rate U.S. $29. surface rate to other countries U.S. $57; air rate U.S. $85. All subscriptions begin with the current issue. Please allow up to 5-6 weeks for first copy.

Just for your TRS-80® COLOR TDP System 10 Dragon-32 MC-10

CoCo Community

We are compiling a list of Color Computer Clubs because of the many requests we have received. The CoCo Clubs may want to exchange newsletters, share ideas for topics of discussion at monthly meetings, etc.

Please let us know if we have omitted any clubs and send us complete, up-to-date addresses. Also, please notify us if you wish to add or delete any names on this list. Send your information to:

Color Computer Clubs
c/o Suzanne Kurowsky
the Rainbow
9529 U.S. Highway 42
P.O. Box 209
Prospect, KY 40059

ALASKA

Alaska Color Computer Users' Group, Rick McDannel, 430C Beluga Ave., Ft. Richardson, AK 99505 phone (907) 428-0392

ARIZONA

Tucson 6809 CoCo Club, Steve Parkman, 902 S. Kolb Rd. Tucson, AZ 85710 phone (602) 747-8233

ARKANSAS

Central Arkansas CoCo Club, Melinda Braslovsky, 1203 Erving Rdg., LP Cabot, AR 72023 phone (501) 982-8854

CALIFORNIA

Los Angeles CoCo Users' Group, Mark Randall, 2227 Canyon Rd., Arcadia, CA 91006 phone (213) 355-6111

Joe Bennett, 1169 Florida Street, Imperial Beach, CA 92023, phone (474-6213

Los Angeles-Wilshire Color Computer Users' Group, c/o Norm Wolfe, 269 S. Lafayette Park Pl., Los Angeles, CA 90057

Ventura County Color Computer Club (VC4), c/o Pete Lyall, Oxnard Pubic Library, 214 "C" Street, Oxnard, CA 93030, phone (805) 984-1842 or DATA (805) 984-1842

COLORADO

Lowry Microcomputer Club/CoCo Users' Group, Jerry D. Surritte, 2249 Moline St., Aurora, CO 80010, phone (303) 343-3273

FLORIDA

C.C. Club of Sarasota, Ernie Bontrager, 4047 Bee Ridge Rd., Sarasota, FL 33582, phone (813) 921-7510

Broward CoCo Club, Sue Spahn, 11950 N.W. 29th Manor, Sunrise, FL 33323, phone (305) 741-4737

ILLINOIS

Town & Country CoCo Kenwood, Chicago, IL phone (312) 493-3748

Motorola Microcomputer Club, Steve Adler, Pres., 1301 Algonquin Rd., Schaumburg, IL 60196, phone (312) 576-3044

KENTUCKY

Roger Idstrom, 2603 Garden Lake Lane, Louisvile, KY 40220, phone (502) 491-1853

Lo-CoCo Club Liason, 2820 Del Rio Place #27, Louisville, KY 40220, phone (502) 458-0649

LOCO-COCO, c/o Mike Standefer, 3141 Doreen Way, Louisville, KY 40220, phone (502) 458-6990

MASSACHUSETTS

New England C.C. Users Group, Christopher E. Sweet, P.O. Box 255, Harvard, MA 01451, phone (617) 456-8291

Massachusetts CoCo Club, Jason Rahaim, Spring St., Lunenberg, MA 01462, phone (617) 582-6514

MICHIGAN

Greg Miller, P.O. Box 365, Haslett, MI

Midland C.C. Club, Neil Drake, 709 Coolidge, Midland, MI 48640, phone (517) 631-2939

Michiana Coco Club, Clay Howe, 310 S. Jefferson St., Sturgis, MI 49091, phone (616) 651-4248

MISSISSIPPI

Singing River C.C. Club, Jerry P. Lowe, Sr., 2500 Fairly Road, Gautier, MS 39553

MISSOURI

North County 80 Group, Tom Vogel, 12 Ville Donna Ct., Hazelwood, MO 63042, phone (314) 739-4078

MONTANA

Billings C.C. Club, Jayne Kenyon, 4306 Phillip, Billings, MT 59101

NEW JERSEY

Bug 80 Users' Group, George R. Miller, Jr., Box 62, Glen Gardner, NJ 08826

Loco CoCo Club, Bud Lavin, 73B Wavercrest Ave., Winfield Pk., NJ 07036

NEW YORK

Adirondack CoCo Club, Bill Edwards, Box 365, Bolton Landing, NY 12814, phone (518) 644-9927

Kings Byte CoCo Club, Morty Libowitz, 1063 East 84th St., Brooklyn, NY 11236, phone (212) 763-4233

C.C. Club of Central N.Y., Joseph Short, 248 S. Fourth Avenue, Ilion, NY 11357, phone (315) 895-7730

NORTH CAROLINA

TRS-80 Users' Group of Charlotte, Bill Hardin, 6613 Summerlin Pl., Charlotte, NC 28226, phone (704) 542-9959

Raleigh Color Computer Club, David Roper, P.O. Box 681, Garner, NC 27529

OHIO

CoCo Club of Youngstown, Timothy McFadden, P.O. Box 478, Canfield, OH 44406, phone (216) 788-4218

Cincinnati TRS-80 Users Group, R.A. White, 44 Dow Court, Fairfield, OH 45014

Paul Selig, 20734 Stanford Ave., Fairview Park, OH 44126, phone 333-2920

Miami Valley CoCo Club, R. Douglas Wales, Pres., 2065 Le Feure Rd., Troy, OH 45373

Dayton CoCo Users' Group, Joseph P. Evans, 609 Applehill Dr., W. Carrollton, OH 45449

OREGON

Willamette Valley CoCo Users, Brian James, P.O. Box 11468, Eugene, OR 97440, phone (503) 687-9286

PENNSYLAVANIA

Jerry Behler, Penn-Jersey Color Computer Club, 1231 Walnut St., Allentown, PA 18102, phone (215) 253-1236

SOUTH CAROLINA

Midlands 80 Computer Club, Robert Ross, P.O. Box 7594, Columbia, SC 29202, phone (803) 776-4361

Metropolitan Greenville CoCo Club, Ed Lowe, P.O. Box 6, Gray Court, SC 29645, phone (803) 876-3928

TENNESSEE

Chattanooga CoCo Club, Jim Perkins/Jim Cox, P.O. Box 9825, Chattanooga, TN 37412, phone (615) 870-2439

Memphis Color Computer Users' Group, Ben Barton, Pres., 4903 Warrington Rd., Memphis, TN 38118, phone (901) 795-7075 or 362-5945

TEXAS

CoCo User Group, David Karam, 1809 Dexter, Austin, TX 78704, phone (512) 442-6317

UTAH

Ogden CoCo, Kathy Rush, 4535 S. 2600W. Roy, UT 84067

VIRGINIA

D.C./N.Va. C.C. Club, Jack Darling, 43 Donovan Dr., Alexandria, VA 22306, phone (703) 780-6159

WEST VIRGINIA

Mtn. State CoCo Users Group, Donald G. Barber, Jr., P.O. Box 295, Granville, WV 26534, phone (304) 599-4493

Mil-O-Bar C.C. Club, Jim Lemaster, Milton, WV, phone (304) 743-4752

Blennerhassett CoCo Club, David Greathouse, Rt. 9, Box 119, Parkersburg, WV 26101

WISCONSIN

Southern Wisconsin Coco Club, David C. Buehn, P.O. Box 411, Twin Lakes, WI 53181

CANADA

Medley Computer and Electonics Club, P.O. Box 1267, Medley, Alberta, Canada T4A 2M0

NOVA SCOTIA

Halifax Dartmouth CoCo Users Group, P.O. Box 572, Dartmouth, N.S. Canada, Bay 3Y9, phone (902) 469-3656

ONTARIO

K-W C.C. Club, Peter Karwowski, 23 Hudson Cr., Kitchener, Ontario, Canada N2B 2V7, phone (519) 579-2953

MEXICO

Marcelo Luft, Laja #232, 01900 Mexico D.F., Mexico City, Mexico, phone 5-68-78-75

THESE FINE STORES CARRY THE RAINBOW

The retail stores listed below carry *the Rainbow* on a regular basis and may have other products of interest to Color Computer users. We suggest you patronize those in your area.

ALABAMA

Birmingham	Jefferson News Co.
	Software City
Florence	Anderson News Co.
Madison	Madison Books
Mobile	Computerland
Montgomery	Trade 'N' Books

ALASKA

Anchorage	Radio Shack
Fairbanks	Electronic World
Kotzebue	Arctic Bookstore, Inc.

ARIZONA

Glendale	Road Runner Computer Products
Mesa	Personal Computer Place
Phoenix	Home Brew Computers
	The Computer Shop
	Software City
	Tri-Teck Computers
Scottsdale	Data Concepts
	Softwareland Corp.
Sierra Vista	Livingston Books
Tempe	All Systems Go
	Books Etc.
	Computer Library
Tucson	Anderson News Co.
	Software City
	Software Unlimited
Yuma	Soft Shop

ARKANSAS

Little Rock	Anderson News Co.

CALIFORNIA

Atwater	Electrobrain
Citrus Heights	Software Plus
Downey	The Floppi Disk
El Cajon	Radio Shack
Exton	Software City
Fortuna	R&V Sound
Half Moon Bay	Strawflower Electronics
Hollywood	Levity Distributors
Lancaster	Alpha Computer Equip. Co.
Los Angeles	OPAMP Technical Books
Modesto	Software Mart
Morro Bay	Coast Electronics
National City	Willy's Electronics
Northridge	HW Electronics
Pacific Beach	Pro Am Electronics
Palo Alto	Printers, Inc.
Riverside	Selectronics
Sacramento	North Area News
	Software Center
San Diego	Accolade Distributors
	Computer Dimension
	The Computer Store
	Dimensional Software
	Disney's Electronics
	Radio Shack
Santa Rosa	Software 1st
Southgate	Color Computing
Stockton	Steven Moreno
Sunnyvale	Computer Literacy
Torrance	Softwoire Centre Int.
29 Palms	Adobe Electronics

COLORADO

Aurora	Aurora Newsland
Boider	The Computer Connection
Colorado Springs	Hathaway's Magazines
Denver	Wholelife Dist.

CONNECTICUT

Danbury	Computer Serv. of Danbury
Hartford	The Aetna Life Club Store
Orange	Software City

DELAWARE

Wilmington	Normar, Inc.—The Smoke Shop

DISTRICT OF COLUMBIA

	The Program Store

FLORIDA

Clearwater	Soenen & Wilmoth Books
Cocoa	Space Coast Software
Coconut Creek	Data Base
Ft. Lauderdale	Mike's Electronics Distributor
	Software Connection
Kissimmee	Radio Shack & Elec. Hut
Longwood	Adventure International Store
Miami	Computer Image
	Joe's News
	Micro Byte
	The News Rack
Orlando	The Alamo
	All Systems Go
	Software Unlimited
Panama City	Computer Systems Group
Pensacola	Anderson News Co.
Sarasota	Family Computers
	Software City
South Pasadena	Poling Place
Stuart	Caribbean Engineering Corp.
Tallahassee	Anderson News Co.
Tampa	Software City
	Software Store
	Sound Trader & Computer Center
Winter Park	Independent Computer Systems

GEORGIA

Atlanta	Chips, Inc.
	Guild News Agency
	Hands On Computer
	Software Atlanta
Columbus	Muscogee News Co.
Marietta	Act One Video

IDAHO

Boise	Magnum Computer Products
Pochtello	Wizerdz

ILLINOIS

Arlington Heights	Software City
Aurora	Kroch's & Brentano's
Champaign	Book Market
Chicago	B. Dalton Booksellers
	N. Walbash St.
	West Jackson St.
	Bob's in Newtown
	Bob's News Emporium
	Bob's Rogers Park
	Book Market
	East Cedar
	North Cicero
	West Diversey
	Chas. Levy Circulating Co.
	E.B. Garcia & Associates
	Guild Books & Periodicals
	Kroch's & Brentano's
	South Walbash
	West Jackson
	516 N. Michigan
	835 N. Michigan
	Parkway Drugs
	Parkwest Books
	Sandmeyer's Bookstore
	Univ. of Chicago Bookstore
	Univ. of Illinois Bookstore
	Videomat, Inc.
Danville	Book Market
DeKalb	Appletree Computers
Evanston	Chicago-Main News
Lisle	Book Nook
Newton	Bills TV Radio Shack
Oak Brook	Kroch's & Brentano's
Oak Park	B.I.E.S. Systems
	Kroch's & Brentano's
Peoria	Book Market
	Illinois News Service
Schaumberg	Data Domain
	Kroch's & Brentano's
Skokie	Kroch's & Brentano's
S. Holland	Abacus Computers
Taylorville	Wilson's Radio Shack
Wheaton	Skyline Marketing
Wheeling	North Shore Distributors
Wood Dale	Pace-Micro Software Centers

INDIANA

Crown Point	Unelex Software
Elkhart	Elkhard City News
Garrett	Finn News Agency
Greenwood	The Computer Experience
Indianapolis	A-Computer Store
	Bookland, Inc.
	Game Preserve
	Indiana News
Jasper	Computer Store
	Elex Mart
Lawrenceburg	Bauer Electronics
	Computer Services
Madison	Arcs Office Supplies
Martinsville	Radio Shack
New Haven	Advanced Color Software
Scottsburg	Radio Shack of Scottsburg
Walbash	Mitting's Electronics

IOWA

Bettendorf	Cosmos Computers
Davenport	Software City

KANSAS

Hays	Gulliver's Inc.
Junction City	Ill #'s Stereo
Kansas City	Special Computer Support
Topeka	Palmer News, Inc.
	Town Crier of Topeka, Inc.
Wichita	Amateur Radio Equipment Co.
	Lloyd's Radio

KENTUCKY

Benton	Anderson Electronics
Crestwood	Begley Drugs
Harrodsburg	M & W Electronics
Hopkinsville	Hobby Shop
Louisville	The Computer Store
Paducah	Radia Shack
Paintsville	Gus-Stan Enterprises
Pikeville	Gus-Stan Enterprises

LOUISIANA

Baton Rouge	Acme Book Co.
Shreveport	Computer SOS
Slidell	Radio Shack 7181

MAINE

E. Wilton	O'brian's Electronic Svc. Ctr.
Jay	F.M. Electronics
South Portland	Portland News Co.
Waterboro	Radio Shack

MARYLAND

Baltimore	Micro Computer Center
	The Program Store
Kensington	The Program Store
Rockville	The Software Store

MASSACHUSETTS

Cambridge	Ninis Corner, Inc.
	Out Of Town News
	The Program Store
Dalton	Blue Max, Inc.

Danvers	Microcon Computer & Software Store
Littleton	Computer Plus
Mansfield	Software Shop
Woburn	Microcon Softwarecenters
MICHIGAN	
Ann Arbor	Community News Center
Brooklyn	Weatherwax Radio Shack
Dearborn	DSL Computer Products
Durand	Robbins Electronics
Fenton	Tri-County Electronics Radio Shack
Flint	"I O"
Kalamazoo	John Rollins
Lansing	The Alternate Source
Lapeer	Computer Corner
	User Friendly
Livonia	Level IV Products
Muskegon	The Eight Bit Corner
Novi	MI Software Dist., Inc.
Owosso	C/O Computer Systems
	Hi-Fi Audio Co.
Perry	Perry Oil & Gas
Pontiac	Computer Shack
Rochester	Rochester Book Center
Roseville	New Horizons
St. Johns	Clinton Electronics
Southfield	Software City
Sterling Heights	Sterling Book Center
Utica	Byte By Byte
MINNESOTA	
Brooklyn Center	Gopher Hole
Minneapolis	Read-More News
MISSOURI	
Kansas City	Midwest CoCo Systems
St. Louis	Magic World
	Software Centre
Springfield	January Books
University City	Final Edition
NEBRASKA	
Lincoln	Hobby Town
Omaha	Computers & Components
NEVADA	
Las Vegas	Hurley Electronics
Spark	Computer House
NEW HAMPSHIRE	
Milford	New England Periodical Serv.
Peterborough	Radio Shack
Portsmouth	Portsmouth Computers
West Lebanon	Verham News Corp.
NEW JERSEY	
Bergenfield	Software City
Cedar Knolls	Village Computer & Software
Cherry Hill	Software City
Clinton	Micro World II
Eatontown	The Program Store
E. Windsor	Software 'n' Suds
Englishtown	Software City
Fairlawn	Software City
Fairview	Software City
Freehold	Radio Shack
Lawrenceville	Micro Con Software Center
Linwood	Software City
Midland Park	Software City
Montvale	Software City
Nahwah	Computer Discount of N.J.
Pennsville	Dave's Elect. Radio Shack
Pinebrook	Software City
Red Bank	Software City
River Edge	Software City
Rockaway	Software Station
Summit	Software City
Teaneck	Software City
Wayne	Wayne Software
NEW MEXICO	
Albuquerque	Page One Newsstand
	Salt of the Earth
Artesia	Radio Shack F756
NEW YORK	
Amherst	Buffalo Teknologies
E. Rochester	The Software Shop
East Syracuse	The Data'Phile Co.
Fairport	Software City
Great Neck	Software City
Johnson City	Unicorn Electronics
Melville	Imperial News
Mt. Kisco	Discount Personal Comp. Sftw. Sys.
New York	Barnes & Noble—Sales Annex
	Big Apple Software Co.
	Coliseum Books
	The Computer Center
	Eastern Newsstand
	Grand Central Station, Track 37
	200 Park Ave., (Pan Am #1)
	55 Water Street
	World Trade Center #2
	First Stop News
	Idle Hours Bookstore
	International Smoke Shop
	Jonli Smoke
	Penn Book
	State News
	Usercom Systems, Inc.
	Walden Books
N. White Plains	Software City
Rochester	Village Green
	World Wide News
Syracuse	Frank Hogg Laboratory
Williamsville	Computer Resource
Woodhaven	Spectrum Projects
NORTH CAROLINA	
Havlock	Computer Concerns
Hickory	Papers & Paperback
Rocky Mount	Southern Software
Salisbury	The Memory Bank
Winston-Salem	K & S News Stand
NORTH DAKOTA	
Fargo	Computer Associates
OHIO	
Canton	Little Professor Book Center
Cincinnati	Softwarehouse
Cleveland	Jerry's Computer
Columbus	Software City
	The Program Store
Coshocton	Utopia Software
Dayton	Data Link
	Wilke News
Fairborn	News-Readers
Lakewood	Lakewood International News
Lima	Brunner News Agency
	Edu-Caterers
Lorain	Computer Network
Mansfield	PCLEAR 80
Mayfield Heights	Programs Unlimited
	Software City
Miamisburg	Wilke News
Niles	Computer Showcase
Philadelphia	Little Professor Book Center
Toledo	Leo's Book & Wine Shop
	Reitz Electronics
Westerville	Home Computer Store
OKLAHOMA	
Duncan	Computers 'N' Stuff
Muskogee	Radio Shack
Oklahoma City	Merit Micro Software
	Software Plus
Stillwater	D. Data
Tulsa	The Computer Store
Woodward	H & H Electronics
OREGON	
Eugene	Libra Books
Grant's Pass	L & R Electronics
Medford	John's News Stand
Portland	80-Plus
Roseburg	Rainy Day Bookshop
PENNSYLVANIA	
Allison Park	Software City
Altoona	Newborn Enterprises
Avondale	Perry's News Shop
Exton	Software City
Greensburg	The Program Store
Malvern	Personal Software
Philadelphia	City Software Center
	Newsy
Phoenixville	Stevens Radio Shack
Pittsburgh	All-Pro Souveniers
Pleasant Hills	Pitt Computer & Software
Plymouth Meeting	The Program Store
Tunkhannock	The Donna Comm. Co.
Whitehall	Software City
Williamsport	Shade Tree
York	The Computer Center of York
RHODE ISLAND	
Warwick	Software Center of Rhode Island
	Software Connection
SOUTH CAROLINA	
Beaufort	Data Byte Computer Center
Greenville	Palmetto News Co.
Hilton Head	Megatron Corporation
N.Charleston	The Green Dragon
Union	Fleming's Electronics
TENNESSEE	
Chattanooga	Anderson News Co.
	Chattanooga Choo-Choo
Knoxville	Anderson News Co.
	First Byte Computer Co.
Memphis	Computer Center
	Tobacco Corner Newsroom
Nashville	Campus Computer Corp.
	Mills Book Store
	Misko's Book Store
Smyrna	Delker Electronics
TEXAS	
Austin	Austin News Agency, Inc.
	Capitol Microcomputers
	Software & Things
Beeville	Bee Electronics
Conroe	Crouchet Electronics
Dallas	Dallas Computer Center
	Software Concepts
Denton	Audio Concepts Unlimited
Elgin	The Homing Pigeon
Ft. Worth	RFI Electronics
	Software Terminal
Houston	MicroSolutions
	Recycle Computers
	Software, Etc.
Irving	Software Access
San Antonio	Coles Computer Center
UTAH	
Salt Lake City	Quality Technology
VIRGINIA	
Alexandria	Alonso Book & Periodical
Falls Church	The Program Store
Gafton	Electronics Marketing
Richmond	Software City
WASHINGTON	
Bellevue	Software City
Renton	Data Borne
Richland	C & J Electronics Computer Center
Seattle	Adams News Co., Inc.
Spokane	Bits, Bytes & Nibbles
Tacoma	Nybbles 'N Bytes
Tukwila	Programs Plus
Vancouver	Northwest Effect
Wenatchee	Software Etc.
WEST VIRGINIA	
Elkins	Binary Tree
Kingwood	Appalachian Computing
Parkersburg	Valley News Service
WISCONSIN	
Appleton	Badger Periodicals
Cudahy	Cudahy News & Hobby
Janesville	Book World
Milwaukee	Book Tree
	Booked Solid
	Booked Solid II
	Campus Corner
	Computer Software Center
	Harvey Schwartz Bookshop
	Prospect News & Magazines
	Univ. of Wisconsin Bookshop
Wauwatosa	Acorn Computer Systems
WYOMING	
Casper	The Computer Store
AUSTRALIA:	
SYDNEY	
Kingsford	Paris Radio Electronics
CANADA:	
ALBERTA	
Athabasca	McLeans Ltd.
Barrhead	Barrhead Sound 1982 Ltd.
Brooks	Double "D" A.S.C. Radio Shack
Calgary	Imperial Computer Ltd.
	Rainbow Software Services
Camrose	Radio Shack
Claresholm	Radio Shack
Coaldale	Coaldale Sight & Sound
Drayton Valley	Longard Electronics
Edmonton	CMD Micro
	Kelly Software Distributors
Edson	Radio Shack
Fairview	D.N.R. Furniture & TV
Fox Creek	Fox City Color & Sound
Ft. Macleod	Radio Shack
Grande Cache	The Stereo Hut
Grande Centre	The Book Nook
Grande Prairie	Northern Computer Service
Hanna	Technics TV Ltd.
Leduc	Radio Shack
Lloydminster	Radio Shack
Peace River	Radio Shack
Pincher Creek	Thornton & Son's
Red Deer	Computerworld
St. Paul	Tele-Logic
Taber	Pynewood Sight & Sound
Valleyview	Don's Radio Shop
Vermilion	Photocraft Vermilion Ltd.
Wetaskiwin	Radio Shack
BRITISH COLUMBIA	
Burnaby	Compulit
Ft. St. John	Ken Dawson
N. Vancouver	Microwest Distributors
Victoria	International Software
MANITOBA	
Winnipeg	J & J Electronics Ltd.
NOVA SCOTIA	
Dartmouth	Sector Software
Halifax	Atlantic News
ONTARIO	
Bowmanville	Bowmanville Audio Vision LTD
Etobicoke	NEPCOM
Hamilton	Galls Book World
Kincardine	Prism Software
Kingston	T.M. Computers
London	Multi-Mag
Ottawa	National News Co.Ltd.
Toronto	Intercity News Co., Ltd.
QUEBEC	
Sherbrooke	Soc. De Loc
SASKATCHEWAN	
Regina	George Glass
Saskatoon	Computertime
YUKON	
Whitehorse	Big Byte Computer Services
	H & H Electronics
PUERTO RICO	
San Juan	Software City

Also available at all Waldenbooks and selected B.Dalton Bookseller stores in the United States and Canada.

ADVERTISER'S INDEX

We encourage you to patronize our advertisers—all of whom support the TRS-80 Color and TDP System-100 computers. We will appreciate your mentioning the RAINBOW when you contact these firms.