COLOR COMPUTER NEWS

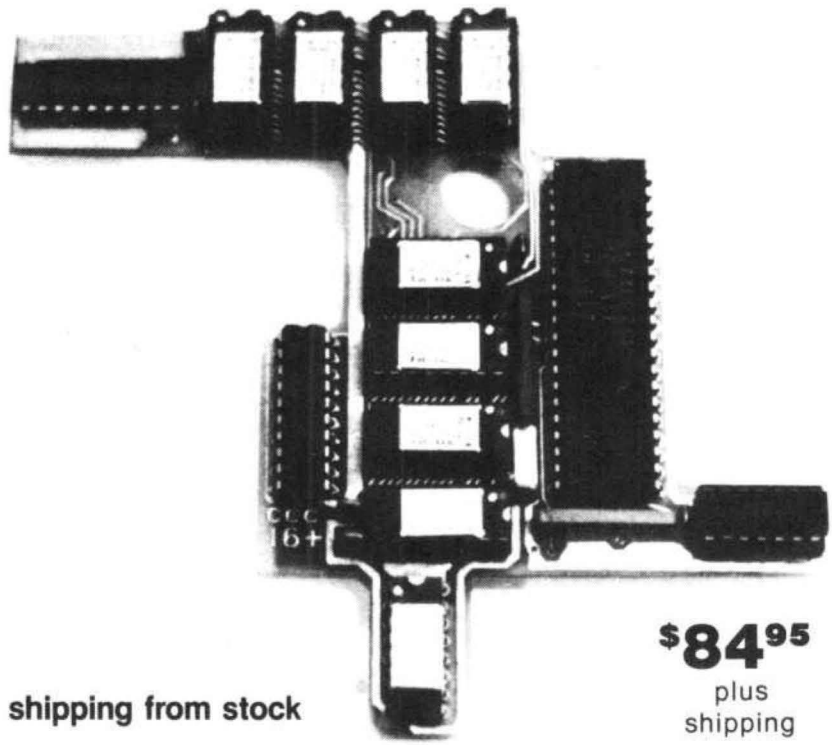
REMarkable Software
P.O. BOX 1192
MUSKEGON, MI 49443

BULK RATE
U.S. POSTAGE
PAID
MUSKEGON, MI
PERMIT #220

16 PLUS

MEMORY EXPANSION FROM 16K TO 32K FOR THE

COLOR COMPUTER



shipping from stock

$84.95 plus shipping

- No soldering! Easy to plug in!
- Fits under RF shield — complete instructions included
- Allows video display (graphics) to reside anywhere in the 32K memory
- Call or write for our complete catalog of Color Computer products!

COMPUTERWARE®

Box 668 • Encinitas, CA 92024

6809 Specialists (714) 436-3512

Computerware is a trademark of Computerware.

by Bill Sias

We've finally finished moving into our new office. Our new telephone number is (616) 759-0682 and the address is 2370 Henry Street, Muskegon, MI 49441. The old box number is still valid and should be used for subscriptions and whatever. Phone hours are Tuesday through Friday from 9:30 am to 3:30 pm and Saturday from 9:00 am until noon, all times are Eastern Daylight Time. I hope the hours aren't inconvient for anyone but we needed to limit the number of phone hours so that we could get some work done. If you are ever in Muskegon feel free to stop by.

I've received several requests from Hams wanting to know what my call was and what bands I work. Well I'm WD8JUJ and I work primarily 15 and 20 meters almost exclusively CW. However from this date forward I can frequently (bad pun intended) be found at the following frequencies and times:

Tuesday	9:00 - 10:00 EDT	14.300
Thursday	9:00 - 10:00 EDT	14.050 or 14.225
Saturday	7:00 - 8:45 EDT	3.050
Most any time		146.940 rpt

We've been talking about a CC net and perhaps one of these can work into one. I'm also open to schedules, just send a letter with date, time and frequency. Be warned I'm a DXer and rare countries take presidence.

While on the subject of Ham Radio, I've been using the MFJ CW-1200 and it's great. It can dig out weak signals almost as well as my ears can. It also has almost no trouble with QRM when used with a good CW filter. It and Clay Abrams software are an unbeatable combination, although I'm currently working on my own software to add things like auto-incrementing serial numbers for contests and an automatic log book.

It's really winter in Michigan now. We just had our first real storm and the electricity was out for a few days. I hadn't realized how computer dependant REMarkable was until we had to live without them for a couple of days. We spent the time well however and I think we're caught up with the back issue orders now. If your back issue came hand addressed please forgive the shaky handwriting, we were all shivering at the time.

We just ran into a hassle with an author, and while I don't want to embarrass anyone I'm sure that now he'll agree that this point needs mentioning. Please when you send an article in don't send it to more than one magazine at a time. This fellow sent it to half a dozen and it was accepted by at least three of them. He then had the problem of writing each of them back and telling them that another had accepted it. He also ended up making less money than he could have because of this. I was, however, pleased (and shocked) to find out that our rates are competative with some of the bigger magazines.

Speaking of articles, we still don't have a sufficient back log in the file and they really aren't coming in fast enough. I am pleased to report that the vast majority of articles received to date are extremely good. Some suggestions and requests: please send all program listings on cassette, even a short listing can be botched up on this end and we don't have the time to test every possible combination of parameters to be sure we haven't let an error slip in, it would also be helpful if the program (BASIC) was saved in ASCII format. What about you "newcomers", I keep getting letters telling me to run more "beginner" articles but I can't run them until you folks write them. Let's go gang!

You have by now noticed the change in format. The double columns allow us to pack about 1/3 more information per page. There are several reasons for the change, the first is "more bang for the buck", the second is that we needed the practice laying out the double columns in preparation for the changes to come down in the near future (are you curious?).

The program listings in this issue were made two ways. Some were done with TEXTPRO (a word processor from CER-COMP) and the others were done with a program sent to me by David Bodnar (our Education Contributing Editor). TEXTPRO worked out quite well until we hit some listings without spaces. Word Processors space out lines by looking for the end of a word next the end of your specified line length and some of the programs had "words" that were 100 characters and more. After adding some spaces it worked extremely well. TEXTPRO has some automatic commands that allowed me to add the spaces easily without manually checking and editing each line.

Speaking of Word Processors; WNET in Boston is showing TELEWRITER in a future production. I haven't any more details than that but as I get more information I'll pass it along. Fred Simon (of the Cincinnati Enquirer) also reviewed TELEWRITER in a recent column.

Dear Bill
Have recently received my copy of CCN for Nov/Dec. I'm pleased to see it is continuing to expand and that you will be going monthly.
To help offset Bob Vaughan's tale of woe I would like to commend a different supplier:- Nelson Software Systems, P.O. Box 19096, Minneapolis, MN 55419. Encouraged by the articles in the July/Aug issue I looked around for a supplier to provide me with a kit to upgrade my 4K (really only 2 and bit) to 16K. I picked this supplier after studying the advertisers in CCN and 80 Microcomputing purely at random based on price and the advertising copy. The kit arrived promptly, the instructions although not extensive proved to be adequate (with the background from your articles) and after 30 anxious minutes and some heavy breathing I had a working 16K machine. The test program proved out and its been running fine since. I suppose you pays your money and takes your chances.
Incidentally I tried to get a copy of the Service manual before I did this but you would never believe the problems my local Radio Shack dealer went through to get it. The Catalog No 26-3001/3002 did not seem acceptable (per the July/Aug issue) and we nearly ended up buying two more Color Computers. I believe Part No. 874-9295 may be a better bet.
Now that I have 16K I was hoping to get into more of the programs in CCN but I notice these are tending to be mostly written in Extended BASIC. I'm not at all certain I want Extended Basic completely especially with the hassle of PCLEAR, etc. I would like some of the subroutines available though and this brings me to two points:-
a). Is anyone selling a chip to fit in the Extended BASIC chip socket with alternate programs/ routines?
b) In CCN could you please indicate which programs are in Extended BASIC? It is not always immediately obvious and to be halfway through typing a program and then realize it won't work is frustrating. Perhaps a few notes on possible conversion procedures if practical could also be added.
Lastly based on a review in the last issue I went ahead and bought the SECS program from DataSoft. Basically it works pretty well especially the editing utility. However your reviewer was correct in saying that the documentation for the graphics is sketchy. For a rank beginner who had not had access to your review article it would have been disastrous. So far I have ironed out a few bugs and written to DataSoft about others. If you are interested in an update when I get the answers and a possible program (either game or graphics) then I will try to oblige.
Yours very truely
Ian L. Smith
P.O. Box 395
Johnsonville, SC 29555

But on the other hand!

Dear Mr. Sias
I wish to "Dart" and "Laurel" some firms advertising products for the Color Computer. The "Laurel" goes to both Computerware and The Micro Works. Their products work, and work well, and are shipped virtually immediately. An example (typical) is eight days from order mailed to product received. Our "Dart" victim should deliver like this!
My "Dart" goes to Nelson Software Systems (Box 19096, Minneapolis, MN 55409). I ordered their Adventure game Pack on July 9, 1981, enclosing a check, which they promptly deposited. A second note, sent 18 October elicited no response.I guess that is what I get for ordering from firms that don't advertise in the Color Computer News!
I'd recommend that all color Computer News readers save their money and avoid Nelson Software Systems.
Sincerely Yours
Charles L. Perrin
P.S. I am forwarding a copy of this to Nelson Software System, so that they can write you with any comments that they may have.

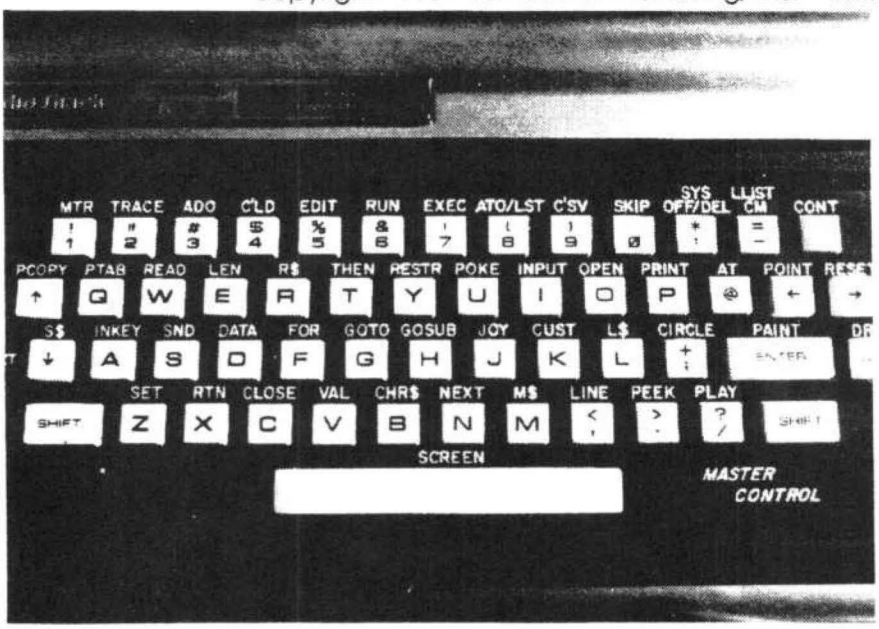
Dear Ian
It's nice to hear a good report about a supplier. Most of the people in this field are good honest folks and it's nice to hear people say so. You are right about the manual, we used the Catalog number on the front of the book not realizing that it was the part number for the product the service manual was about. Third, yes I'll try to label Basic programs as to which Basic they are for. Some times I don't realize because most arrive on tape. Lastly, I would be delighted to publish an article/program that uses SECS. It's an excellent product with less than excellent documentation (I've seen alot of those).

Gentlemen
On Page 6 of Volume 4, the screen dump program does not work. In line 130 the data is directed to memory starting at 51500 - this is within the address area of the external ROM packs.

For Your Color Computer

MASTER CONTROL

Copyright ©1981 Soft Sector Marketing, Inc. – Written by A. Swartz



Requries 16-32K

1. 50 preprogrammed command keys. Standard and Extended command.
2. Direct control of motor, trace, and audio from keyboard.
3. Automatic line numbering.
4. Programmable Custom Key.
5. Direct Run Button.
6. Keyboard overlay for easy program use.
7. Easy entry of entire commands into computer.

Load Master Control into your machine then either type in a BASIC program or load one in from tape to edit. Cuts programming time by 50% or more **$24.95**

Coming for Christmas!

COLOR BONANZA

50 Programs for the Color Computer.
Less than $1.00 a program!

Some 4K, some 16K, some extended BASIC, some for non-extended.

Games - Personal Programs - Utility Programs

List **$49.95**

Expected shipping date December 1 or before.
Any prepaid orders received before that date will pay only **$39.95**

CONCENTRATION & CONNECT UP 4

Twin Pack - 16K Extended

$14.95

Expected Shipping Date - December 1 or before

From Other Vendors

Space Invaders, Spectral Associates	**$21.95**
Space War*, Spectral Associates, *req. Joy Sticks	**$21.95**
Meteoroids*, Spectral Associates, *req. Joy Sticks	**$21.95**
Battle Fleet, Spectral Associates	**$14.95**
Space Traders, Spectral Associates	**$14.95**
Software Development System Rom Pack	**$89.95**
C Bug Monitor	**$29.95**
Disassembler (Source Generator)	**$49.95**
Color Computer News (Latest Issue)	**$2.50**

Animated Hangman

He winks... he blinks... he almost lives!
An outstanding game for the whole family!

Non Extended BASIC
Requires 16K
$12.95

C.O.D. – certified check, M.O. or cash only. Most orders shipped next day. All orders must have shipping included. Please add 2% or $2.50, whichever is higher for shipping. Michigan residents, please add 4% tax. Add extra $1.50 for C.O.D. Personal checks take 3 weeks to clear. Out of the country orders add $10.00 extra shipping.
*TRS-80 is a product of Radio Shack, div. of the Tandy Corp.

SSM SOFT SECTOR MARKETING, INCORPORATED
6250 Middlebelt • Garden City, Michigan 48135
Order Line **800-521-6504**
Michigan Orders & Questions **313-425-4020**

The American Express® Card
Don't leave home without it.™

MasterCard VISA

Following is my screen dump program that I always insert at power up. Trying to do complex engineering programs and watching the TV receiver can be most annoying. With this program the PRINT#-2 and LLIST are no longer required. The printer and video Terminal display full 80 character lines.

```
5 'THIS PROGRAM SETS THE COMPUTER FOR
CONTINUOUS OUTPUT TO THE RS232 JACK
6 'TO PROVIDE CONTINUOUS READOUT ON A
VIDEO TERMINAL OR PRINTER WHEN WRITING,
7 'LISTING, EDITING OR RUNNING A PROGRAM.
10 FOR X=1 TO 15
20 READ A
30 POKE 1007+X,A
40 NEXT X
50 POKE 359,126
60 POKE 360,3
70 POKE 361,240
80 DATA 52,22,198,254,215,111
90 DATA 190,160,2,173,3,15,111
95 DATA 53,150
```

This will remain in memory until power down.

Very truely yours

Francis Sherwood

Dear Francis,

Thanks for sharing your printer routine. The routine in the last issue does work with the Exatron CCI because it puts RAM in addresses from $C000 up.

Dear Sirs;

For three months our club has been trying to find a program that will input and output to the RS232 port via the modem without interrupting Basic. We want to hook between two and ten Color Computers together and still have full control of the computer eg. if you type SOUND 100,1 on the keyboard of this computer it will SOUND 100,1 .. what we want to do is this. Suppose we have two Color Computers hooked together over the phone lines. I want to be able to type the Sound 100,1 on my computer and have both my computer and his obey the command.

Here is how we want it to work.

1. It must act as an extention of Basic, not interrupt it.
2. It must accept input from the RS232 port and treat it exactly the same as input from the keyboard.

3. Anything that is inputted from the keyboard or sent to the screen by the computer must also be outputed to the RS232 port.
If anyone can help please write to me. PLEASE!!
Please put your phone number in your ads. I put off getting your mag for three months because I couldn't get you by phone to subscribe (I hate to write letters) and I usually subscribe to mags by phone. Anyway, I don't know when my subscription runs out, but when it does please renew it and send me the bill.
You have the best magazine for color Computer users on the market.
How about a program or command series to operate Radio Shack's remote appliance controller from Basic without calling a machine language subroutine? Speaking of which, what good is this unit? I mean what I bought the unit for is to turn lights on and off. Now suppose you are writing a 200 line program and you are at line 110, it starts getting dark so you want to turn the lights on. Who would save the program you are working on, load the R.S. controller program, turn the lights on then reload the original program and continue. Thats stupid. If you had a simple command like PRINT UUUUUXU or something like that it would be great. As is I have never found a use for it.

Keep up the good work.
Yours Truely,
Ron Garrett
2101 East Main St
Henderson, TX 75652

Dear Ron,
Chaining the computers together sounds like a great idea. We could then play multiplayer games between computers. They do it all the time with other computer and there must be someone somewhere doing it already with the Color Computer. Radio Shack's controller was really designed for the Models 1 & 3. Both of those have hardware clocks in them and it would be a fairly simple task to read the clocks and turn the lights on/off or whatever. You could write a machine language program that would work as a software clock and program it to do the functions that you need. If the program were interrupt driven you could run it as a background task and not have to save your program but run them both at the same time.

Dear Bill,
Thanks for the chat on Friday night. Murphy's law struck again. When I got home the latest and

greatest issue of CCN was waiting for me. I of course spent the night reading it, and will take it with me on my trip today.
As the case has been with Prior issues, this issue is better, and I am glad you are going monthly as that will give me something to look forward to every four to six weeks, depending on the speed (or lack of) the Postal Service. I am also enclosing a check for $2.50 to order the first issue. I would like to keep a complete library of CCN.
I will be ordering the Exatron CCI as I agree with your comments in our phone conversation and in the latest CCN issue. Your article indicated that you wrote some additional commands for the CCI. Will you be writing any articles on how to do that? The changing of CAT to DIR sounds easy, but adding commands sounds like fun and will be very useful. I have added commands to SSM's MASTER CONTROL and would love to add commands to the CCI. I would also like to "PATCH" programs to read/write to diskette as you did to the Personal Finance ROM Pack.
Again thanks for another great CCN issue and I hope to write a review for SSM's MASTER CONTROL.
Yours truely
Tom Flanagan
32 Arthur Street
R.F.D. 1 Box 248
Peekskill, NY 10566
P.S. Good luck with the MICROSOFT COBOL bug we spoke about.

Murphy's Law was rampant. The Cobol bug did bite, but I knew it would thanks to your warning. My accounting program now keeps track of where it is on the disk rather than depending on Cobol to issue an error. The problem I told you about was in fact caused by the bug.
I'm really pleased that #4 was waiting for you at home. We mailed 2 days late last issue but it seemed to get delivered late to a lot of folks. The problem that we get into is with new subscribers, we have to mail by third class because first class mail on the magazines would at least double the subscription price. Many new subscribers worry about the money they sent because the first issue doesn't arrive within a few weeks. As you have seen the post office doesn't even deliver it in less than two to three weeks and we can't mail unless we have the minimum requirement. The only solution I have come up with is that we mail the current issue on the 24th of each month. For example, we mailed the November/December issue on the 26th of October and the 26th of November. Now that we are a monthly publication we'll be mailing each issue just once and it will become a back issue the day after mailing. If anyone has a better idea I'm quite open for suggestions.

Dear Bill,
I would like to contact other CC users in my area to see if they are interested in getting together locally. Those interested can contact me.
Sincerely
Rick Ruggles
5529 Corkhill Dr
Dayton, OH 45424
233-6715

Dear Sirs(or Bill):
I'm delighted with your magazine, CCN. I'd been shopping around and have found some material in 68 Micro Journal, Byte; etc. Some of their material is good, but there's nothing else out there with so much volume of relevant and valuable material.
Question: How do I send you material for an article or a letter in the form of a Color Computer Cassette tape? Of course, I can send a program that's been CSAVE 0. Is there a good standard form for text?
I think you might want to answer this letter in a coming issue. I'm sure the problem has deterred at least a few possible contributors. I started to write a small text processor using CR's at the end of a line, but I ran into trouble on reading back the string under INPUT#-1. I lose the CR's. I'd consider CC Writer or Telewriter, but don't know which to chose.
Very Truly Yours,
Arnold H. Kahn

Dear Bill,
I have just received my first issue of CCN (Nov/Dec). I must say that after reading it, I am looking forward to the three back issues and especially to the 1982 monthly issues. I have sent under separate cover, the additional money to cover the cost of all three back issues sent first class.
I have a couple questions that I hope you will have time to answer. I am still a little green (about the color of the Color Computer display) when it comes to the internal works of the Color Computer so if the following questions seem a little basic, I apologize!

1. The first question is about Vitamin E, that poke instruction that increases the speed of the CPU. This instruction works when I run under Extended Basic but with Disk Extended Basic it will now. The system hangs up. Do you know why? Is there still a way to use the poke when running Disk?
2. The second question is about the Videotex article by Gregory Cegielski in the Nov/Dec issue. I would like to do the reprogramming of Videotex. How can I do the suggested programming if I am not allowed to remove the Videotex RomPack when the computer is on and Videotex has control of the system?
Thank you for any consideration you may give to these questions.
CCN is great!! Please keep it up.
Ronald Beatty

* I'm not certain but the speed POKE (or Vitamin E as it's become famous as (you heard it here first)) will certainly clobber the disk I/O routines because disk is extremely time sensitive. Secondly, the Videotex article was written (and approved) before Videotex became a ROM Pak, yes Radio Shack did sell at least one cassette for the Color Computer.

Dear Bill:
You can see already that I don't have a printer and that I make lots of typos, at least till I get warmed up. Last Spring I bought a 4K Color Computer to see how a computer and I got along. I eagerly opened the Manual and started typing away so I could learn as much as possible before fishing season (Commercial) started in May. My first experience with Tandy's confusion came quickly. No Operation Manual. I finally got one from the store but no help with the many other things I had questions about. Granted they were stupid beginners questions but I guess they weren't so stupid that they didn't get a lot of "I don' know's" out of the Service Rep in Fort Worth. Of course that was all on my phone bill because Alaska is included in the "Foreign Country" or Outside the Continental U.S.A." catagory. I must say that they were helpful and sent me photostats of parts of their second try at the Manual.
Even after that inauspicious beginning I still was impressed with the Color Computer. I plugged along in between working on the Alaska Ferries and as Summer approached my Troller. It became apparent about three quarters the way through the "Getting Started Manual" (maybe I should have specified that earlier)---now, if I had Word Processing and a printer I could go back andstraighten that up---and a few fumbling attempts at producing my own programs (math practice for my twelve year old- disastrous attempts at Home Finance) that 4K was just not enough. And I worked at compression, hard. Byte's article on the Color Computer showed how to put your own chips in to upgrade to 16K so I ordered a set, with full instructions. Lesson number two, watch who you telephone order from. I finally gave up and ordered from an established company, after talking to them. About a week before fishing both sets arrived, the first without instructions, not even C.O.D. as requested. Just with a bill. Upward to 16k but no time to play with it.
I had decided by that time that the EDIT and RENUM functions plus the graphics (for my youngster) required Extended Basic so I plunked the set on the local dealers desk and told him I would see him in September.
In between all this I discovered an ad for Color Computer News and immediately sent off in time to get Vol. 1 Number 1. The only complaint I have is that it is bi-monthly. I am anxiously awaiting the September-October issue.
What little I have played with the computer since laying up the boat a week ago has reaffirmed my decisions to upgrade and go Extended Basic. I can see that soon I am going to need 32K though. (One can also see that I need to also learn how to compress my writing too). I don't have issue #1 with me on the Ferry (where I am typing this) but you asked for input as to what we desired. MORE---period.
Well, I do have something move specific than that. Since the Color Computer is aimed at the low end, new user market (I think at least)--oh to be able to EDIT that goof--and that most of us aren't too far from thinking more of Sirloin when talking of bytes and the old lady when talking of (a little) bits.
We have about 20 or so Color Computers in Juneau (pop. about 20-25K, total area) but I have not been in contact with any other users to date. Last Spring we started a Users Group (all makes) and we hope to begin again next month. Maybe I will be able to get some input and ideas then. For now, I have to feel that possibly Mr. Dingwall's letter in the Sept. 81 80 Microcomputing, p.12, hits home. Tandy's total lack of support, information, assistance and of course secrets etc, etc, etc, only helps support this response. I really wonder how many of the 20 here are sitting on shelves in the closet.

I purchased the R.S. Finance Pak thinking that I could use it for my fishing business. I only ran a couple of minutes of it in the store and wondered about it but went ahead. It only took me ten minutes at home to discover that I had been doing what it offered for several years by hand and much faster and easier. I took it back and the folks graciously traded it for another program pak. Since my youngster already has the ATARI game machine and bunches of tapes, I did not feel that any of the R.S. game packs would interest him at all so I settled for the Chess pak. This was partially in self defense. I thought I might be able to sneak in some practice while he was at school and possibly be able to last more than a dozen moves against him. Well, it was at least somewhat successful, he grabbed onto the thing and had it running and half beaten before he read the instructions. The problem now is that he won't do his other homework. Back to the books (Chess anyhow).

I hope you are able to get an advance copy of Color Scripsit and get a review in before it comes on the market in November. I suspect that is asking too much. As you can tell, Word Processing is one of my main interests, I anxiously peer in the ads in Byte, 80 Microcomputing, 68 Micro-Journal; and of course in the reviews in all the magazines, to see what is coming. I see that there is a CC Writer out (ad in 80 Microcomputing Sept 81 p326), a Super Color Writer (same mag. p 184) that sounds too good to be true, plus the several Editors you mentioned on p 17 of Vol 1, #2. I just wonder how much memory will be left (even with 16K) after you load one of these in RAM. I also wonder how furiously someone worked to punch them out and whether they show the hurried issue. As you can tell, I am very suspicious of R.S. and Scripsit since I am reading the De-Bug column too. How do you De-Bug a ROM pak?? Sounds to me like you have what you get.

As soon as the finances permit I want to get a printer. I am entranced by both the Epson 80FT and the Okidata Microline 82. It sort of sounds like an interface is necessary for both, maybe not the Microline. I just don't know enough about the whole area to tell. We don't have enough stores up here to find out anything either. And my phone bill is horrendous enough as it is, so I don't really need to call California etc.

Also of interest is the ad by Exatron on;4, Vol 1 #2. Again I know nothing about what is going on in this area. Is it $298.00 plus the cost of a disk and you are running, extra 32K and all? I don't think so. Of course R.S. advertises Disk for the C.C. at $599.00.

I am sure you can see my point, probably several paragraphs ago. In addition I am interested in learning more about Assembly Language but first I need to get thru both the updated R.S. Basic Manual and the Extended Manual and learn how the machine works. One thing for sure, 80 Microcomputing's Assembly Line and assorted articles lose me in the first paragraph.

Bill, I hope you are a good speed reader. I certainly am filling alot of paper up here.

I think I will close off by saying that I have ordered six months worth of "Chromasette Magazine", with great reservations I might add, and if I come across anything will try to give you a "big dummies review". I know I am much too verbose a writer but maybe you can edit out the garbage and make some use of it. Whatever. I note that one of the magazines I take (about five or six Computer mags) offers guides to reviewers. If you have anything on that order you wish to forward I would be more than happy to digest it. Probably the only advice you wish to offer would be to take about two years of writing courses at my local college. I darn near didn't graduate from Denver University (1966 at 32 yrs., Hotel and Restaurant Management Degree) due to my lousy English abilities.

I also have ordered Spectral Associates "Magic Box". Of course I don't have any Mod I or III tapes to load but figure it may save a bunch of agony in trying to modify the many Mod I programs I find in books and 80 Microcomputing. Might have something to say on that for you too.

Keep up the good work. Do want to see your next issue announcing that you are going monthly soon. Bigger and better and more often (did I hear you say "oh no, please not that"?).

Your avid fan,
William H. Sanford
Box 750
Juneau, Alaska 99802
Chief Purser
Alaska Ferry System

PS Say--I just addressed the envelope. REMarkable Software??? Where are your ads for software for the CC. Wayne Green is forever pushing Instant Software, why can't you. Maybe not enough time, starting up the mag and all.

Well, going backwards, I don't sell CC software for one primary reason. Is it fair for me to ask all these folks for money to advertise their products here and then be their #1 competitor? I

do get the point you are making in that most of the CC owners are new to computing and all of the magazines (including CCN) are too technical. I have two excuses as to why it's extremely difficult to change the situation, as far as CCN is concerned. First, I'm learning that as the magazine grows more of my time is spent being a publisher and less being a writer. Second, I can only print what comes in and very few beginners articles are arriving.

I am enclosing a program which I have written. I have written it myself and any similarity to other programs is purely coincidental. The program is for the radio "HAM" operator and will allow the operator to recall the other station's name by entering the call letters. Additional calls and names can be easily added to the data statements. This program has been a big help to me.

Sincerly,
James Parsons

```
10 REM PROGRAM TO GET NAME FROM CALL
SIGN
20 REM BY JAMES PARSONS, K5ROV
30 REM ENTER CALL & NAME AS DATA LINES
40 CLS
50 CLEAR 250
60 PRINT"NAMES OF HAM OPERATORS
WORKED:"
70 PRINT:PRINT
80 PRINT"ENTER CALL LETTERS OF STATION:"
90 INPUT A$
100 READ B$,C$
110 IF B$="END" AND C$="END" THEN
RESTORE:GOTO 250
120 IF A$=B$ THEN GOSUB 150:RESTORE:GOTO
190
130 GOTO 100
150 FOR X=1 TO 10
160 PRINTB$;"----------";C$
170 NEXT X
180 RETURN
190 PRINT"ANOTHER CALL (Y/N)"
200 C$=INKEY$:IF C$="" THEN 200
210 IF C$="Y" THEN CLS:GOTO 80
220 IF C$="N" THEN PRINT"GOOD DAY":GOTO
260
250 CLS :PRINT"STATION NOT LISTED" :PRINT
:PRINT :GOTO 80
260 END
1000 DATA K5ROV, JIM
1003 DATA WD4FTO, FRANK
```

Making Eduction More Colorful

By David Bodnar

As I mentioned in the last issue, one of the main purposes of this column will be to evaluate software with an eye toward its use by educators. The program that I am about to describe should be of interest to anyone who has to type letters, reports, or even computer magazine columns. It is a Text Editor/Word Processor program that turns the COLOR COMPUTER into a very versitile word processing machine.

For those of you who have never used a word processor a general definition may be in order. In its simplest form a word processor takes any information that you type on the keyboard and sends it to a printer. This one line word processing program fits this simple definition:

```
10 INPUTA$:PRINT#-2,A$:GOTO10
```

This program has some obvious problems and limitations so lets take a close look at a word processor that has very few problems and/or limitations.

COLOR COMPUTER
TEXT EDITOR/WORD PROCESSOR

The program comes to you on a cassette that is accompanied by an extensive (18 page) instruction manual. The program will run on a 16K machine but its full potential is reached when 32K of RAM is available.

Its features include:

- UNRESTRICTED INPUT OF TEXT
- CASSETTE STORAGE OF TEXT
- ADD TO TEXT FROM KEYBOARD OR TAPE
- MERGE TAPE FILES
- DISPLAY TEXT WITH VERTICAL SCROLL
- DISPLAY ANY LINE OF TEXT
- DELETE ANY LINE OF TEXT
- INSERT NEW LINES ANYWHERE
- EDIT ANY LINE
- AUTOMATIC TITLE CENTERING
- ONE KEY INDENT (TAB)
- FIND ANY WORD IN TEXT
- OUTPUT TO PRINT
- PRINT A DRAFT (FULL OR PARTIAL)

After CLOADING (yes it is almost entirely in BASIC) and entering RUN a ">" prompt is displayed. Responding with "S", the START commmand, causes the program to print "FROM KEYBOARD OR TAPE (K/T)" on the screen. Since this is our first run through we have no tape file and respond by entering "K". The screen clears and becomes black with a blue cursor in the upper left corner. When you begin typing you find yourself in the lower case mode. Holding SHIFT while keying a letter key produces UPPER CASE. Just like a typewriter!!. SHIFT ZERO will lock the keyboard in the upper case mode for emphasizing words or titles.

You may now type text up to the memory limits. This is about 200 line on 16K and 500 lines on 32K. The only distraction will occur after typing about 48 lines. A tone will sound and the keyboard will be disabled for several seconds while the program takes care of some chores. ENTER is used only at the end of a paragraph or when you wish to skip down for one or more lines. To escape from the text input mode key the <up arrow> and the system prompt ">"reappears.

After typing the text you have several options. If you wish to view the entire text enter "/" and the text will scroll by. SHIFT @ will pause. If you wish to edit you must first find the line number of the line you wish to edit. The "FIND" command seems to be the easiest way to do this. If you wanted to correct a typographical error such "COMPTUER" with "COMPUTER" respond to the ">" prompt with "F" for "FIND". You will be asked "FIND WHAT?". Enter "COMPTUER" and any and ALL occurrences of "COMPTUER" will be displayed proceded by the line numbers of the line (s) where "COMPTUER" occurs. Now that you know the line number (let's say it is 26) enter "26E", for line 26 EDIT. The screen will clear right below it; respond with "COMPTUER". "INSERT?" will be displayed with "COMPUTER" in place of "COMPTUER". One word of caution. If you wanted to replace "CAR" with "BUS" in this text line: "The carrot was under the car." you would get "The busrot was under the car." To get around this respond to "DELETE?" with "<space> car <space>" and respond to "INSERT?" with "<space> bus <space>". This allows the editor to distinguish between "car" in "carrot" and "car" when it is between spaces.

The text editor also allows you to insert or delete lines anywhere in the text. Simply enter 26D to delete line 26 or 26I to insert a line or lines before line 26. Responding to the ">" prompt with "?" will get a status report that includes the current line number, the number of lines left till the memory is filled, the number of characters typed and the number that may still be entered. To return to entering text simply enter "ADD" and continue typing.

After the text has been entered and edited to your satisfaction you may either send the text to the printer with an "o", output, command or save the text on cassette with an "EXIT" command. I suggest saving to tape first so that an unexpected power failure won't erase several minutes or hours of work.

The "O" command produces an output menu that allows you to change margins, select line spacing, automatically right justify text by adding spaces within lines, break a long text into pages, set the number of lines per page and select the number of copies to be printed.

Before using "O" you may wish to have a full or partial draft of the text printed so that you may "pencil edit". The command "D" will print a full draft complete with line numbers. "PR" will print a partial draft.

One of this program's most versatile features is VARIABLE TEXT. This permits you to insert markers in your text that will be replaced, when sent to the printer, by text that is stored in a separate file. It may sound complicated but we have all seen it used in letters we have received about contests or magazine subscriptions that have our name and address cleverly woven into the letter so we feel that it is personalized. One way it might be used by a teacher would be to send notices to 12 students' parents about their child's progress (or lack of it). The teacher would need 12 nearly identical letters. The only differences might be the addresses, the parent's name, the child's name a pronoun such as him/her and the subject area. Using VARIABLE TEXT a single letter would be written and every time a variable came up a "1" would be entered. After the letter is completed a variable text file is created containing the 12 sets of variables. When printed the 12 letters would be indentical except that the variables would replace the "1" symbols.

The command "M" allows you to move a block of text from one location to another. This would be used if you wanted to relocate a paragraph. You may also duplicate a block of text in another place with the "DUP" command, or delete a block of text with the "D" command.

There are two versions of the program available. One is for computers with 32K of RAM. The other is somewhat limited version for 16K RAM. The 16K version is limited to about 200 lines of text. To give you an idea of how much that is, this column is now on line 198. The 16K version also lacks several other features, VARIABLE TEXT, (MOVE, DUPLICATE AND BLOCK DELETE are) among them. If you purchase the 16K version an owner registration card is included that may be sent in for a free copy of the 32K version of the program when and if you expand your memory to 32K.

The text editor supports the two printers that seem to be used by most Color Computer owners, the R/S Line Printer VII and the Epson MX-80. If you have a different printer the fact that the program is written in BASIC makes it easy to modify. By changing a few lines I even got it to output to a Quick Printer II with surprisingly good results.

As you may have gathered from the comment I made a few paragraphs ago I am using the text editor program to write this column. Even though the Color Computer's small screen size limits its potential as a text editor, this program does a fine job of lifting the CC above those limitations and turns it into a very versatile word processor.

AUTHOR: John Wacle
P.O Box 11224
Pittsburgh, PA 15238

Text Editor Program
$49.95 + $2.00 shipping and handlings

REMarks

Those of you that get Chromasette have by now read Dave's comments about us and Rainbow. We both were unaware of the other at the times our ads were laid out, and I have changed my ad accordingly. For those that don't get Chromasette; we were advertising ourselves as the ONLY totally Color Computer magazine on the market, which at the time was true. Lonnie Falk (Rainbow's Publisher) and I had a long chat on the phone shortly after we discovered each other and agreed that the competition was good for both of us so we exchanged ads.

The "Those Friendly Folks at ...." column is absent this issue. It seems that an ex-White House employee crept in during the night and erased the tape the phone interview was recorded on. The interview with Frank Hogg Labs will be redone and transcribed in time for next issue. The schedule goes as follows: February; Frank Hogg Labs, March; CER-COMP, April; Aardvark, May; Soft Sector Marketing. Let me know who you would like to hear about and I'll try to work them into the schedule.

Last issue many folks got confused by the ad on page 4 and thought it was REMarkable's ad. It was actually from Soft Sector Marketing and those orders that we received were passed along.

DRAW III

by Don Inman

Material for this article is from the book "TRS-80 Color Computer Graphics". Now in publication by Reston Publishing Company, Inc. Ruston, VA 22090

In the two previous issues of Color Computer News, I discussed most of the optional commands that can be used in the DRAW statement of the Color Computer. In this issue I'll introduce the use of substrings within a DRAW statement and then tackle the problem of mixing text and graphics while in a graphics mode.

Substrings

You probably know how a subroutine works in a computer program using BASIC. You can perform a similar feat within the string of a DRAW statement. The X motion command lets you execute a substring of a string.

We have separated some commands from others in previous DRAW statement examples by semicolons. Usually, the semicolon is optional. However, when the substring command X is used, it must be followed by a semicolon. This is true even is the X command is the last command of the string.

Example:

```
150 A$ = "E10;F10;G10;H10"
```

← substring to be used

these semicolons are optional

```
250 DRAW"BM60,90;L30;U30;R30XA$;D30XA$;"
```

these are necessary

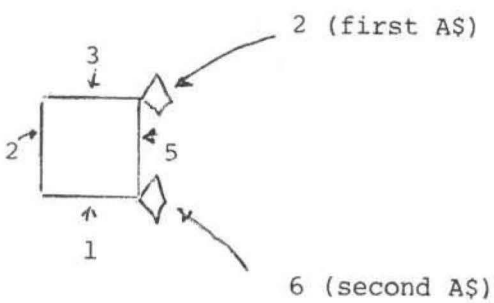
these are optional

With the minimum number of semicolons, the example would be:

```
150 A$ ="E10F10G10H10"

250 DRAW"BM60,90L30U30R30XA$;D30XA$;"
```

only these are necessary

Lines 150 and 250 would draw the segments of the figure in the order incicated.



The example could be used in the following program.

```
100 PMODE 3,1
110 PCLS
120 SCREEN 1,0

130 A$ = "E10F10G10H10"
140 B$ = "H10G10F10E10"

150 DRAW"BM100,80;R30XA$;D30XA$;L30XB$;U30XB$;"

160 FOR X = 2 TO 4
170   PAINT(134,80),X,4
180   FOR W = 1 TO 10: NEXT W
190   PAINT(134,80),1,4: PAINT(134,110),X,4
200   FOR W = 1 TO 10: NEXT W
210   PAINT(134,110),1,4: PAINT(98,110),X,4
220   FOR W = 1 TO 10: NEXT W
230   PAINT(98,110),1,4: PAINT(98,80),X,4
240   FOR W = 1 TO 10: NEXT W
250   PAINT(98,80),1,4
260 NEXT X

270 GOTO 160
```

Two substrings (A$ and B$) are defined at lines 130 and 140 and used to DRAW diamonds at the corners of the square at line 150. Notice that the substrings are all followed by a semicolon, as required. The FOR-NEXT loop (lines 160-160) color and erase the insides of the diamonds in succession. Press the BREAK key to stop the program.

Adding Text to Graphics

One of the first disappointments that I experienced when I started using my Color Computer was the lack of ability to display text and graphics at the same time from BASIC. However, once I began experimenting with the DRAW statement, the solution was obvious. Since text characters can be formed from straight lines, my first approach was to use the DRAW statement to label portions of a graphics display.

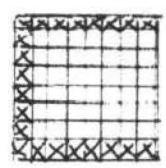
The problem boiled down to the following:

1. Choose a size that would be readable in all graphics modes
2. Choose a uniform way to DRAW the letters
   a) start at the same relative position
   b) end at a position that could be used as a starting point for a succeeding letter
3. Store the letters in some way so that they could easily be found

1. Choosing the Size

It seemed logical to chose a size that could easily be scaled downward, if desirable, by the DRAW statement itself. Therefore, an 8 * 8 grid was used for each letter of the alphabet.

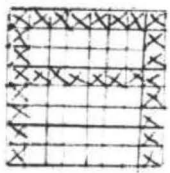
Examples: The letter C



The DRAW statement would include: L8D8R8 or L8U8R8

or ?

The letter A



U8R8D4L8BR8D4 or ???

2. Choosing a Uniform Way to Draw

The method was selected in a purely arbitrary manner but provides consistency. The first decision was to start and end in the lower left corner of the grid.

It was seen, almost immediately, that this should be followed by a balnk move to the right 12 positions. This would put the DRAWing "pen" in position to start a succeeding letter in its lower left corner. It would also provide spacing of four positions between letters. This is demonstrated in the following program.

```
100 PMODE 4,1
110 PCLS
120 SCREEN 1,0

130 CIRCLE(128,96),30

140 'DRAW THE LETTERS
150 DRAW"BM95,148;U8R8BD8L8BR12"
160 DRAW"BU8R8BL4D8BR4L8BR12"
170 DRAW"U8R8D4L8BR4F4BL8BR12"
180 DRAW"U8R8BD8L8BR12"
190 DRAW"U8BR8BD8L8BR12"
200 DRAW"U8R8BD4L8BR8BD4L8BR12"

210 GOTO 210
```

3. Store the Letters

Obviously, this process is time consuming and must be done for each letter unless some way is used to save the shapes. One way to do this is to write a subroutine (or subroutines) made up of the DRAW commands of a separate letter. Each letter could be assigned its own name as:

A$ = "U8R8D4L8BR8D4BL8BR12"
B$ = "U8R6F2D2L8BR8D2G2L6BR12"
etc.

This method uses up a lot of string variables and was discarded.
Instead, an array named L$ was used with appropriate subscripts.
L$(1) = DRAW string for A
L$(2) = DRAW string for B
L$(26) = DRAW string for Z
One additonal element was added to provide a space between words
L$(27) = "BR12"
the final subroutine used was:

```
10000 'CHARACTER SET FOR GRAPHICS
10010 L$(1) = "U8R8D4L8BR8D4BR4"                          A
10020 L$(2) = "U8R6F2D2L8BR8D2G2L6BR12"                   B
10030 L$(3) = "U8R8BD8L8BR12"                             C
10040 L$(4) = "U8R6F2D4G2L6BR12"                          D
10050 L$(5) = "U8R8BD4L8BD4R8BR4"                         E
10060 L$(6) = "U8R8BD4L8BD4BR12"                          F
10070 L$(7) = "U8R8BD4L4BR4D4L8BR12"                      G
10080 L$(8) = "U8BR8D8BU4L8BD4BR12"                       H
10090 L$(9) = "BU8R8BL4D8BL4R8BR4"                        I
10100 L$(10) = "U4BU4BR8D8L8BR12"                         J
10110 L$(11) = "U8BR8G4L4BR4F4BR4"                        K
10120 L$(12) = "U8BD8R8BR4"                               L
10130 L$(13) = "U8F4E4D8BR4"                              M
10140 L$(14) = "U8F8U8BD8BR4"                             N
10150 L$(15) = "U8R8D8L8BR12"                             O
10160 L$(16) = "U8R8D4L8BD4BR12"                          P
10170 L$(17) = "U8R8D8H4BG4R8BR4"                         Q
10180 L$(18) = "U8R8D4L8BR4F4BR4"                         R
10190 L$(19) = "BU4U4R8BD4L8BR8D4L8BR12"                  S
10200 L$(20) = "BU8R8BL4D8BR8"                            T
10210 L$(21) = "U8BR8D8L8BR12"                            U
10220 L$(22) = "BU8D4F4E4U4BD8BR4"                        V
10230 L$(23) = "U8BR8D8H4G4BR12"                          W
10240 L$(24) = "E8BL8F8BR4"                               X
10250 L$(25) = "BU8F4E4BG4D4BR8"                          Y
10260 L$(26) = "BU8R8G8R8BR4"                             Z
10270 L$(27) = "BR12"                                     space
10280 RETURN
```

The subroutine has been saved on tape and is loaded previous to any program in which it is to be used. Here is how it is used with the previous CIRCLE program

```
80 CLEAR 500                    <- clear lots of string space
90 DIM L$(27)                     <- dimension the array

100 PMODE 4,1
110 PCLS
120 SCREEN 1,0

130 CIRCLE(128,96),30

140 GOSUB 10000                 <- assign elements of the array

150 DRAW"BM95,148"+L$(3):L$(9)+L$(18)+L$(3)+L$(12)+L$(5)

160 GOTO 160
```

C I R C L E

Since each character (with spaces) takes 12 horizontal graphics positions, 21 characters will fit on a line if you start at the extreme left of the SCREEN.

An Alphabet Game

Now, how do I remember the numbers of the letters of the alphabet? I decided I would write a practice exercise that turned into what I call the "Alphabet Game". It includes the Color Computer's TIMER function to add some spice.

Ten random letters are shown on the SCREEN (one at a time). You make a guess (even a wild one) as to what the corresponding array subscript for the displayed letter is. Penalties are provided for time and the magnitude of the error in number.

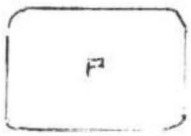
```
10 'ALPHABET GAME

100 CLEAR 500
110 DIM L$(27),Q(10)                  <- You can change Q to fit the
120 CLS                                   the number of turns
130 PMODE 4,1
140 PCLS
150 SCREEN 1,0
160 GOSUB 10000                   <- don't forget to load the alphabet

200 Q=0                           <-  Q keeps track of your score
210 FOR X = 1 TO 10                       10 turns
220   R = RND(26)

230   DRAW "BM120,100"+L$(R)        <-  show letter
240   TIMER = 0                       <- start timing
250   DRAW"BM2,12;U8"+L$(27)          <-  ready for 1st digit
260   A$(1)=INKEY$: IF A$(1)="" THEN 260
270   T1=TIMER: TIMER=0               <- get time of entry, reset timer

280   DRAW"BM2,12;U8BR12D8"           <- ready for 2nd digit
290   A$(2)=INKEY$: IF A$(2)="" THEN 290
300   T2=TIMER                        <- time for second entry

310   N=10*VAL(A$(1))+VAL(A$(2))      <-  calculate guess time
320   E=ABS(R-N)                       <- calculate error
330   T=INT((T1+T2)/6)                <- total time

340   Q(X) = T+20*E                         <- error penalty
350   PCLS
360 NEXT X

400 CLS:PRINT"YOUR SCORES WERE :"
410 FOR X = 1 TO 10
420   Q = Q+Q(X)
430   PRINT"#";X;Q(X)
440 NEXT X

450 PRINT: PRINT"TOTAL SCORE =";Q

500 PRINT:PRINT
510 PRINT "PRESS ANY KEY TO PLAY AGAIN"
520 A$=INKEY$: IF A$="" THEN 520
530 GOTO 140
```

Don't forget, the alphabet subroutine must be loaded. Have fun learning your alphabet all over again. Feel free to modify the game for higher or lower rewards and penalties. Here is a sample of how the program is used.

DRAW III

Two digits must be entered for each guess that you make. If the letter is A, you would enter: 01. If the letter is B, you would enter: 02. If C, enter: 03, ... If Z, enter 26. A prompt is given in the upper left corner so that you can see which digit is expected.

Example: Random number 16 is selected at line 220:

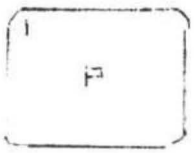


L$(16)

P is the 16th letter. Entry would be: 16

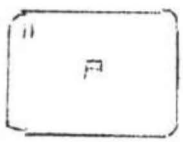
ready for 1st digit

ENTER: 1



ready for 2nd digit

ENTER: 6



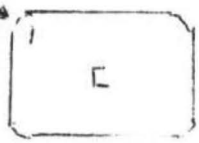
Example: Random number 3 is selected at line 220:

L$(3)

C is the 3rd letter. Entry would be: 03

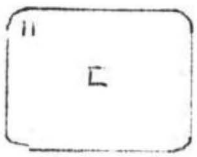
ready for 1st digit

ENTER: 0



ready for 2nd digit

ENTER: 3



After ten turns, your score will be displayed.

YE OLDE HANGMAN

by Richard Masek
Wickliffe, OH

This program provides sound and animation to the age old parlor game of guessing an unknown word. Play begins as the computer searches its data file for a random word. The unknown word is then presented to the player as a series of question marks within a box. Each question mark represents one letter. A visual prompt begins flashing to "guess a letter" and a 15 second timer begins a countdown. When a letter is selected, the computer compares the letter guessed with each letter in the unknown word. If the letter is wrong, the building of the gallows and victim commences. If the letter is a correct guess, then each place the letter is used in the unknown word is now shown within the box.

There are ten parts to the gallows and victim. These parts can be built on every guess or as an option, only built on wrong guesses. This option usually allows several extra chances to guess the word.

If the unknown word is correctly guessed before the victim is completed, the victim jumps down to the base of the gallows and shows his rejoicing by doing a brisk dance to the tune "Happy days are here again." On the other hand, if the word is not guessed in time, the victim drops down to a taunt noose and slowly swings back and forth to the "Funeral march".

One other interesting aspect of the program is the opportunity to select easier words. This allows a youngster's spelling words to be singled out from among a much more challenging selection.

The visual format is divided into two sections. The right one third of the screen is used for the gallows and victim. The left part then keeps a running score of how many victims have been set free and how many victims are now "waiting for the resurrection". In addition, the percentage of correct guesses to total guesses is shown. Below the score, the unknown word is shown within a box. The length of the box changes with the length of the unknown word. The word length is limited to ten characters without causing distortion of the video presentation. Underneath the boxed-in unknown word is kept a running list of the letters guessed. If the correct word is never guessed, it will appear beneath the swinging feet of the victim.

The program.

00-60 Introduction and screen format for score
70-99 String assignments for gallows & victim
100-138 Selecting the random unknown word and boxing-it in
150-170 Request player to guess a letter
175-199 Check if the guessed letter is in the word
200-256 Build the gallows and victim
260-299 Play death march and display message
350-390 Play happy days and display message
400-425 Request if another game desired, initialize data
500-510 Print strings for swinging the victim when hung
520-569 Print strings for victim dancing when set free
900- Data file for unknown words. Each data set includes a difficulty code(1 or 2) and four unknown words

Variables

OG-Correct Guess, 1 if the letter guessed is correct
D-Difficulty, code in data set
EW-Easier word, 1 if the words in the data set are easier
LG$-Letter guessed by the player
NC-Number correct guesses made by player on current word
NG-Number of guesses made by player on current word
NW-Number of wrong guesses made by player on current word
TC-Total correct guesses made by player
TF-Total freed victims
TG-Total guesses made by player
TH-Total hung victims
WA-Wrong answers (used to determine part of gallows/victim)
WD$-unknown word
WL-Word length (number of letters in unknown word)
WX$-Modified unknown word

Miscellaneous

40 TG=1, set to eliminate division by zero
100/101 Unknown word is randomly selected from sets of four
150/151 15 second counter
158/160 Flashing prompt to guess a letter
170 If no guess in 15 seconds, a wrong guess is attributed
126/176 The unknown word is modified by the computer to eliminate a correct letter from being selected over and over.

```
 0005 CLS
 0006 AA$ =CHR$ (128)
 0010 PRINT@7,
"****************"
 0011 PRINT@39, "*YE OLDE
HANGMAN*"
```

```
 0012 PRINT@71,
"******************"
 0013 PRINT@131, "A RICHARD
MASEK PRODUCTION"
 0014 PRINT@194, "GALLOWS &
VICTIM CAN BE BUILT"
 0015 PRINT@226, "EITHER ON
EVERY GUESS (E) OR"
 0016 PRINT@258, "ONLY ON WRONG
GUESSES (W)"
 0017 PRINT@292, "'W' ALLOWS
MORE GUESSES!":  PRINT@330,  AA$
; "type";  AA$ ; "e";  AA$ ;
"or";  AA$ ; "w"; AA$
 0018 Q$ =INKEY$ : IF Q$ =""
GOTO 18
 0019 IF Q$ ="W" THEN WA=1
 0020 PRINT@386, "DO YOU WANT
EASIER WORDS ONLY?":  PRINT@426,
AA$ ; "type"AA$ "y"AA$ "or"AA$
"n"AA$
 0021 Q$ =INKEY$ : IF Q$ =""
GOTO 21
 0022 IF Q$ ="Y" THEN EW=1
 0049 TG=1
 0050 CLS
 0055 PRINT@1,
"******************"
 0056 PRINT@33, "*YE OLDE
HANGMAN*"
 0057 PRINT@65,
"******************"
 0058 PRINT@97, "#FREE #HUNG
SCORE"
 0059 PRINT@130, TF; :
PRINT@137, TH;
 0060 PRINT@141,
INT((100*TC/TG)); "%";
 0070 AA$ =CHR$ (128)
 0071 AB$ =AA$ +AA$
 0072 AC$ =AB$ +AB$ +AB$
 0073 AD$ =CHR$ (58)
 0074 AE$ =CHR$ (129)
 0075 AF$ =CHR$ (130)
 0076 AG$ =CHR$ (111)
 0077 AH$ =CHR$ (140)
 0078 AI$ =CHR$ (142)
 0079 AJ$ =CHR$ (135)
 0080 AK$ =CHR$ (141)
 0081 AL$ =CHR$ (139)
 0082 AM$ =CHR$ (46)
 0083 AN$ =CHR$ (137)
 0084 AO$ =CHR$ (134)
 0085 AP$ =CHR$ (133)
 0086 AQ$ =CHR$ (138)
 0087 AR$ =CHR$ (120)
 0088 AS$ =CHR$ (143)
 0089 AT$ =AS$ +AS$
 0090 AU$ =AH$ +AH$ +AH$ +AH$
 0091 BJ$ =AA$ +CHR$ (117)+AA$
 0092 BB$ =AE$ +AA$ +AF$
 0093 AV$ =CHR$ (131)
 0095 CA$ ="GUESS A LETTER    "
 0096 CB$ ="guess"+AA$ +"a"+AA$
+"letter"
 0097 CC$ ="COME ON!! I CAN'T"
 0098 CD$ ="HANG AROUND ALL DAY"
 0099 CE$ ="                    "
 0100 WD=RND(30)
 0101 XW=RND(4)
 0105 FOR W=1 TO WD
 0110 READ D, XA$ , XB$ , XC$ ,
XD$ : NEXT W
 0111 IF EW=1 AND D<>1 THEN
RESTORE: GOTO 100
 0115 IF XW=1 THEN WD$ =XA$
 0116 IF XW=2 THEN WD$ =XB$
 0117 IF XW=3 THEN WD$ =XC$
 0118 IF XW=4 THEN WD$ =XD$
 0125 WL=LEN(WD$ )
 0126 WX$ =WD$
 0130 PRINT@162, AH$
 0131 PRINT@194, AA$ :
PRINT@226, AV$
 0133 FOR L=1 TO WL+1
 0134 PRINT@162+L, AH$ ; :
PRINT@194+L, "?";
 0135 PRINT@226+L, AV$ ;
 0136 NEXT L
 0138 PRINT@193+L, AA$
 0150 FOR L=1 TO 15
 0151 FOR M=1 TO 40
 0152 LG$ =INKEY$ : IF LG$
<>""THEN175
 0153 IF L>12 THEN PRINT@290,
CC$ ; : PRINT@320, CD$ ; : GOTO
161
 0158 IF M<20 THEN PRINT@290,
CA$ ;
 0160 IF M>20 THEN PRINT@290,
CB$ ;
 0161 NEXT M
 0165 NEXT L
 0170 LG$ ="?"
 0175 FOR L=1 TO WL
 0176 IF LG$ =MID$ (WX$ , L, 1)
THEN PRINT@194+L, LG$ ; : CG=1:
NC=NC+1: WX$ =LEFT$ (WX$ ,
L-1)+CHR$ (128)+RIGHT$ (WX$ ,
WL-L)
 0177 NEXT L
 0178 NG=NG+1
 0180 IF NG<9 THEN
PRINT@416+2*NG, LG$ ; : GOTO 184
```

```
 0181 IF NG<17 THEN
PRINT@432+2*NG, LG$ ; : GOTO 184
 0182 PRINT@450+2*NG, LG$ ;
 0184 IF NC=WL THEN TF=TF+1:
GOTO 350
 0185 IF CG=1 THEN CG=0: PLAY
"O3; L16; C; E; G; L8; O4; C;
L16; O3; G; L4; O4; C": IF WA=1
GOTO 150 ELSE GOTO 200
 0199 PLAY"O1; F; C"
 0200 NW=NW+1
 0205 ON NW.GOTO 210, 215, 220,
225, 230, 235, 240, 245, 250,
255
 0210 PRINT@467, AC$ AC$ ; :
PRINT@499, AC$ AC$ ;
 0211 GOTO 300
 0215 FOR L=0 TO 11
 0216 PRINT@83+32*L, AB$ ; :
NEXT L
 0217 GOTO 300
 0220 PRINT@51, AB$ AC$ ;
 0221 GOTO 300
 0225 PRINT@90, AD$ ; :
PRINT@122, AD$
 0226 GOTO 300
 0230 PRINT@153, AE$ AA$ AF$ ;
 0231 PRINT@185, AA$ AG$ AA$ ;
 0232 GOTO 300
 0235 FOR L=0 TO 3
 0236 PRINT@218+32*L, AA$ ; :
NEXT L
 0237 GOTO 300
 0240 PRINT@216, AI$ AH$ ;
 0241 PRINT@247, AI$ AJ$ ;
 0242 GOTO 300
 0245 PRINT@219, AH$ AK$ ;
 0246 PRINT@252, AL$ AK$ ;
 0247 GOTO 300
 0250 PRINT@345, AN$ ;
 0251 PRINT@376, AN$ ;
 0252 GOTO 300
 0255 PRINT@347, AO$
 0256 PRINT@380, AO$ ;
 0260 PRINT@290, "CHOKE GASP
CHOKE!"; : PRINT@438, WD$ ;
 0261 PRINT@320, "BYEBYE CRUEL
WORLD!";
 0264 TH=TH+1
 0265 BA$ =AS$ +AM$ +AS$
 0266 BB$ =AS$ +AE$ +AA$ +AF$
+AS$
 0267 BC$ =AS$ +AA$ +AR$ +AA$
+AS$
 0268 BD$ =AT$ +AI$ +AA$ +AK$
+AS$
 0269 BE$ =AT$ +AP$ +AA$ +AQ$
+AT$
 0271 BG$ =AS$ +AA$ +AS$
 0272 BH$ =AS$ +AQ$ +AS$ +AP$
+AS$
 0275 GOSUB 500
 0276 FOR X=1TO5: NEXT X: C=-1:
GOSUB 500
 0277 SOUND 19, 10: C=0: GOSUB
500
 0278 FOR X=1TO5: NEXT X: C=1:
GOSUB 500
 0279 SOUND19, 10: C=0: GOSUB
500
 0280 SOUND19, 3: C=-1: GOSUB
500
 0281 SOUND19, 10: C=0: GOSUB500
 0282 FOR X=1TO5: NEXT X: C=1:
GOSUB500
 0283 SOUND58, 10: C=0: GOSUB500
 0284 SOUND45, 3: C=-1: GOSUB500
 0285 SOUND45, 10: C=0: GOSUB500
 0286 SOUND19, 3: C=1: GOSUB500
 0287 SOUND19, 10: C=0: GOSUB500
 0288 SOUND5, 3: C=-1: GOSUB500
 0289 SOUND19, 15: C=0: GOSUB500
 0299 GOTO 400
 0300 PRINT@320, CE$ ;
 0305 GOTO 150
 0350 PRINT@290, "I'M SAVED
ALIVE  ";
 0351 PRINT@320, "THANKYOU!
THANKYOU!";
 0355 PRINT@130, TF; :
PRINT@137, TH;
 0358 FOR A=0 TO 7:
PRINT@150+32*A, AS$ : NEXT A
 0361 GOSUB 560
 0362 SOUND 89, 5
 0363 GOSUB 540
 0364 SOUND 125, 2
 0365 GOSUB 520
 0366 SOUND 125, 9
 0367 GOSUB 540
 0368 SOUND 147, 5
 0369 GOSUB 560
 0370 SOUND 147, 5
 0371 GOSUB 540
 0372 SOUND 176, 2
 0373 GOSUB 520
 0374 SOUND 176, 9
 0375 GOSUB 540
 0376 SOUND 193, 5
 0377 GOSUB 560
 0378 SOUND 193, 5
 0379 GOSUB 540
 0380 SOUND 176, 2
 0381 GOSUB 520
 0382 SOUND 176, 9
 0383 GOSUB 540
```

```
 0384 SOUND 147, 5
 0385 GOSUB 560
 0386 SOUND 147, 5
 0387 GOSUB 540
 0388 SOUND 125, 3
 0389 GOSUB 520
 0390 SOUND 125, 9
 0400 PRINT@352, "***ANOTHER
GAME?***";
 0401 TC=TC+NC: IF TG=1 THEN
TG=TG+NG-1: GOTO 403
 0402 TG=TG+NG
 0403 PRINT@137, TH; :
PRINT@141, INT(100*TC/TG)"%";
 0404 Q$ =INKEY$ : IF Q$ =""
GOTO 404
 0405 IF Q$ ="Y" THEN 420
 0406 IF Q$ ="N" THEN 409
 0407 PRINT@352, AB$ AB$
"type"AA$ "y"AA$ "or"AA$ "n"AB$
AB$ ;
 0408 GOTO 404
 0409 CLS: PRINT@226, "THANKS
FOR PLAYING. HANG IN!"
 0410 PRINT@416, " "
 0411 END
 0420 RESTORE
 0421 NW=0: CG=0: NG=0: NC=0
 0425 GOTO 50
 0500 PRINT@153+C, BA$
 0501 PRINT@184+C, BB$
 0502 PRINT@216+C, BC$
 0503 PRINT@247+C, BD$
 0504 PRINT@279+C, BE$
 0505 PRINT@311+C, BE$
 0506 PRINT@345+C, BG$
 0507 PRINT@376+C, BH$
 0508 PRINT@408+C, BH$
 0510 RETURN
 0520 PRINT@185, AS$
 0521 PRINT@217, BB$
 0522 PRINT@246, AT$ AP$ BJ$ AQ$
 0523 PRINT@280, AO$ AH$ AA$ AH$
AN$
 0524 PRINT@313, AS$ AA$
 0525 PRINT@345, AS$ AA$
 0526 PRINT@377, AS$ AA$
 0527 PRINT@408, AS$ AN$ AS$ AO$
 0528 PRINT@440, AN$ AS$ AT$ AO$
 0529 RETURN
 0540 PRINT@185, BB$
 0541 PRINT@217, BJ$
 0542 PRINT@246, AU$ AA$ AU$
 0543 PRINT@280, AT$ AA$
 0544 PRINT@313, AS$ AA$
 0545 PRINT@345, AS$ AA$
 0546 PRINT@377, AN$ AS$ AO$
 0547 PRINT@408, AI$ AJ$ AS$ AL$
AK$
 0548 PRINT@440, AS$
 0549 RETURN
 0560 PRINT@185, AS$
 0561 PRINT@217, BB$
 0562 PRINT@246, AT$ AS$ BJ$
 0563 PRINT@280, AS$ AI$ AA$ AK$
 0564 PRINT@313, AP$ AA$ AQ$
 0565 PRINT@345, AP$ AA$ AQ$
 0566 PRINT@377, AS$ AA$
 0567 PRINT@408, AS$ AQ$ AS$ AP$
 0568 PRINT@440, AS$ AQ$ AS$ AP$
 0569 RETURN
 0900 DATA 1, THE, SIS, MAN, IT
 0901 DATA 1, SEES, ANN, MIT,
MAT
 0902 DATA 1, WILL, WE, WITH,
SAM
 0903 DATA 1, SEE, NAN, SITS,
THIS
 0904 DATA 1, SAT, AT, SID, CAT
 0905 DATA 1, HAT, FAT, MAT, RAT
 0906 DATA 1, TODD, ADAM, MOM,
DAD
 0907 DATA 1, AMY, CAR, FAR,
STAR
 0908 DATA2, RADIO, SHACK, LOVE,
MOST
 0909 DATA2, COLOR, COMPUTER,
RECORDER, BIBLE
 0910 DATA2, HAPPY, SAD, FLAT,
TRAMP
 0911 DATA2, BICYCLE, FLOAT,
BUCKLE, LOVE
 0912 DATA3, HELP, LOST, BOSS,
FLOSS
 0913 DATA2, TOWER, FLOWER,
POWER, SHOWER
 0914 DATA2, FRIEND, BENCH,
HEART, START
 0915 DATA2, CLOCK, ROCK, STOCK,
SHIRT
 0916 DATA2, SHOES, BRUISE,
MOUSE, CLUES
 0917 DATA2, PURPLE, WHITE, RED,
BLUE
 0918 DATA2, ORANGE, BLACK,
BROWN, VIOLET
 0919 DATA2, HOUSE, BLOUSE,
LEMONADE, RANCID
 0920 DATA2, DOORBELL, DUMBBELL,
BRONZE, ZEBRA
 0921 DATA2, IRON, STEEL, BRASS,
COPPER
 0922 DATA2, SWITCH, RECORD,
BATTERY, TONE
```

BERSERK

Jim Plank

Ok, I might as well clear my conscience right now and confess that I'm not a genuine 8 bit, hexadecimal loving computer programmer. In fact, about a year ago I would have thought that a stringy floppy was one of those ties people wear at square dances. Oh sure, I've taken a couple of classes and can throw the bull about CPUs, RAMs, and ROMs around. Deep down though, I know why I like messing around with these micros, and it sure isn't number crunchin.

What I am is a video freak, an electronic game junky. Shameless I know but true. This sad tale began years ago when hours of my mis-spent youth were thrown away on pinball and SAM II, a pre-video shoot-em-up. As I look back now, I realize my down fall was Pong. My hand still shakes a little as I think back to that night I sat down in complete innocence, well almost complete, at the paddle controller of that demon game. The bar was dimly lit and the single eye of the TV (I didn't know CRT then) glowed back devilishly from the gloom. Hours later, with a cramped hand, and empty wallet and glazed eyes I staggered out into the street. I was hooked.

My addiction has carried me through the years to every low down video arcade in town, my pocket crammed with quarters, my palms sweaty. I took to hiding hand-held games in dark corners at the school where I work so that I could take a few hits between classes or on my prep. This would keep me going until I got home.

At home awaited my Atari game machine. My poor wife thought that at last I would be satisfied. No more would I be lost for hours when we went shopping at a mall with an arcade. An addict can't be controlled that easily, however. I began spending the grocery money on game cartridges. She pleaded, to no avail, I had to have just one more game. "this is the last one," I would promise but I knew it was a lie. Sadly my young sons became addicted too and our evenings at home became orgys of Space War and Laser Blast.

Finally, about thirty cartridges later, the Atari high began to wear off, and I began to look around for new thrills. I checked out the new Mattel machine but it seemed only a little more potent then the old faithful Atari. I needed a quantum jump to keep up.

While taking some classes necessary to finance my habit, I ran across microcomputers. Aha! Here was the ultimate in game machines, more powerful than a speeding Atari; able to leap into high res graphics in a single bound. I might even be able to write my own games,(still a touch of innocence left you see).

After suffering the pangs of withdrawal for several months, while I saved and investigated the market, I decided the best machine I could get without selling the house too was the CC And that about brings things up to date, I might go into my harrowing tale of novice programming and the search for software some other time.

All this self-abasement leads me to my main purpose. Yes, there is a purpose to this article. I may not be a true computer person but I have hard won credentials in video games.
Now the Review!

Berserk

There has been a definate shortage of high quality software for the CC, stuff that really takes advantage of the capabilities of this machine and challanges a true video game fanatic. Ron Krebs has certainly been doing his part to close the software gap. First he's given us two really great adventures and now a true classic arcade game, Berserk.

A real shoot em up, in the arcades, it's known as Berzerk by Stern. The intrepid gamester is given three humanoids armed with laser pistols. One at a time he maneuvers them from the left side of the screen into a room. The walls are electrified so it's death if you touch them. In the room are a number of sentry robots, looking rather like malignent trash cans with feet; their legs move, their heads turn as they seek to destroy the invader, you. From their eyes shoot powerful blasts of energy, as they begin their mad dance around the room, you stop and move the arm of your stand in; pressing the fire button you pick them off but as you do so the survivors move more quickly and fire faster. Wait! What's this? Out from a door way bounces what appears to be a leering refuge from a packman game. Your laser has no effect on it. Because you've watched a team mate chomped to death by this mobile stop sign, you decide to flee to another room. Out another door way you go. The screen scrolls to a new and different shaped room. This room too is crawling with robots. True, some of them seem a little retarded as they destroy each other in their haste, but every room is lousy with them. You run. You fire and you dodge on and on, from room to room until the inevitable happens and your last man is zapped. Your score for burning down all those "Bots" is a respectable 3800 not quite the 5000 needed to get an extra man but you can always make another run and maybe this time...

BERSERK

The game plays great, the animation is super, and the graphics are a real treat. The color and sound are also first class. I'd like to know how some of those sounds were done. I can't remember hearing them on my CC before.

This is the first personal video game that I've seen that seems to have almost all the features of its big brother in the arcade. After a close comparison, the only differences were: 1) the size of the rooms, the arcade game seemed slightly larger; 2) the arcade had a voice that boomed out "intruder alert," etc; and 3) the robots in the CC version were better with even more detail than their arcade counterparts.

Mark Data has produced a real jewel with Berserk, one that you won't play for a week and then banish to the tape file to be accidentally erased.

So Mr. Krebs (sounds like junky's alias to me) who ever you are, I loved the adventures but please just one more arcade type, please, I'm down to my last good tape and my hand is beginning to shake.

YE OLDE HANGMAN continued

```
 0923 DATA2, CONTROL, DIGITAL,
ANALOG, COUNTER
 0924 DATA2, CANAL, DUCK,
MARBLE, EVERY
 0925 DATA2, YOUTH, MASTER,
PRIME, SPECIMEN
 0926 DATA2, SINGLE, SKIN,
CAUSTIC, COLOR
 0927 DATA2, WARNING, INFLATION,
SECURITY, SOCIAL
 0928 DATA2, ATTITUDE, CONTRACT,
AVERAGE, OBJECTION
 0929 DATA2, TENNIS, RACKET,
SMASH, VOLLEY
 0930 DATA2, MEDICAL,
LABORATORY, MONSTOR, ANALYSIS
```

ACTION GAMES

The fastest growing producer of computer games for your 6809 has the products you have waited for!!

NEW! ARCADE GAMES FOR THE COLOR COMPUTER

CAVE HUNTER

COLOR BERSERK

Fast paced action • Super Hi-Res Graphics
Dynamite sound effects • Runs in 16K of memory
These games will astonish you with their Detail and Quality.
They set a standard for others to follow.

— ADVENTURES —

Calixto Island • The Black Sanctum

Highly acclaimed by reviewers • Challenging situations
Fast, efficient machine language • Runs in 16K of memory
Save game in progress

Adventures on 5¼ TSC FLEX disc (specify 6800 or 6809)	ea. $24.95
Both adventures on single disc	$39.95
Adventures for color computer	ea. $19.95
Color Berserk for color computer	ea. $24.95
Cave Hunter for color computer	ea. $24.95

Shipped prepaid in continental U.S. California residents, please add 6% tax.

VISA

— MORE COMING SOON —

WE HONOR master charge THE INTERBANK CARD

MARK DATA PRODUCTS

23802 Barquilla, Mission Viejo, CA 92691 • (714) 768-1551

TRS 80 IS A TRADEMARK OF TANDY CORP.

PAC ATTACK
by Richard White

Every user from Captain '80 on is a program tester. When you or I play a new game program we automatically evaluate it and tell our friends. Computerware tells me that they not only test programs themselves but retain other programmers to test them as well before the program goes on the market. They had better because if it's a game program and if I buy it, my nine year old and his friends will give it the ultimate test. And I can measure the success of the program by how frequently I can come home to find two or three testers playing. This game ranks right up with Spectral's Color Invaders as one of the best two games in the house.

In fact I have some concern about how much service one should expect from Radio Shack joysticks before failure. Seemingly, the joystick (one is required for Pac Attack) is in constant motion for hours. This has been the first real joystick hardware test we have conducted as Color Invaders is best played with the arrow keys. By the way, the Color Computer keyboard is wonderously durable considering the continual pounding it has taken running Invaders for five months.

Now my testers are interested primarily in one thing, excitement. A reasonably challenging game with enough variation so as to require strategy and quick response will generally do the job. Simply said, this is action and Pac Attack has plenty. The objective is to control the pac person to run through the "maze" gobbling up pellets at 5 points apiece. The "rare earth muggers" (there are three) come after the pac person with gobbling intentions themselves. But, for a short period after eating an energizer, the pac person can gobble the foes scoring 200 for the first, 400 for the second, 600 for the third, and 800 for all additional on that one energizer. The current record is five on one helping. Of course, if the pac person gets to the mugger a tad too late, the eater becomes the eatee with moans from both the TV and the player. When the screen is cleaned of pellets and energizers, a fresh rack appears, but you get no bonuses.

At the top of the screen the score and the best score for that loading are kept. Some dreamer provided for six digit scores. The first person that makes it deserves to be in a book of records and may end up in a padded cell. There are three difficulty levels, easy, hard, and tuff which are more than sufficient. So far the testers only play tuff when they feel suicidal urges.

This is my first Computerware item, it certainly will not be the last.

MASTER CONTROL

by Theodore P. Hasenstaub
17401 Dartmouth Ave
Cleveland, OH 44111

Being relatively new to computers and programming, it became quickly evident, that I spent an enourmous amount of time, typing programs into my Color Computer, leaving me little time to enjoy the vast capabilities that it had to offer. Drum Roll Please, ///// enter, MASTER CONTROL, by Alan Schwartz, and distributed by Soft Sector Marketing Inc.

MASTER CONTROL is a machine language program for the 16K or 32K Color Computer, enabling the programmer to spend less time typing and more time enjoying. This program can be placed anywhere in the computers memory, although it is set to load into the top RAM. It only takes up 1100 bytes of memory! You load it by clearing some memory in the top RAM (CLEAR 25,15104, in a 16K machine). CLOADM"MASTER", is the tape loading procedure. After it loads, you get an OK prompt, type EXEC, press ENTER, you are now in control, with, MASTER CONTROL.

The self adhesive keyboard overlay which comes with MASTER CONTROL, is easily placed over the keyboard. This provides you with 50 preprogrammed commands. Although this program will run in a 16K standard machine, some of the commands require Extended BASIC in order to function; Paint, Draw, Circle, etc. The first thing that you notice on your screen, after loading, is that the normal cursor has been replaced with a flashing # sign. To use any of the command keys, you first press the down arrow key, then press the A key (which has the command INKEY printed above it). Presto INKEY$ appears on the screen. You then continue on, typing in the rest of the program line. If at any time you wish to disengage MASTER CONTROL (one reason is that it will still activate, when an input command is used in Basic), simply press the down arrow key, then press the *. To re-enter the program, type EXEC, and then press ENTER.

Another feature which MASTER CONTROL offers is, automatic line numbering. This feature will start numbering lines at a preset number, and automatically increase the next line number, after pressing ENTER. To activate this feature, you must first press down arrow key, then press the (. The numbering will start with line number 10, and increase every line number following by 10, every time you press the ENTER key. You can also set your own starting and increment numbers by pressing the down arrow key, then pressing the % key. Your screen will display an inverse video # prompt, which allows you to enter the new numbers. First, enter your starting number, press ENTER (if your number has 5 digits, you don't have to press the ENTER key), next, enter the number by which you want the lines to be increased, then press ENTER. If you make a mistake, or wish to change the numbers, press CLEAR, and re-enter the desired numbers.

Still another feature included in MASTER CONTROL, is a programmable CUSTOM key. This enables you to have a key to use for groups of statements, or commands, that you are using more than once in a program. The CUSTOM key is activated by pressing the down arrow key, then pressing the K key. To change the contents that this holds, so that you may enter your own statements or commands, press the down arrow key, then press the & key. The screen will again display an inverse video #. This enables you to enter your new data. After entering data, press BREAK.

It actually takes very little time to be able to use the different features of MASTER CONTROL. If you have felt as I have, that you've been spending too much time typing, and not enough time enjoying all your computer has to offer, then MASTER CONTROL, at an extremely reasonable price of $24.95, may be the solution to your typing blues.

instead of a printer; it doesn't automatically ouput a linefeed after carriage return. While I was having difficulty using the SDS80C with the terminal printer, I contacted Bob Lentz and Andy Phelps of Micro Works. Their help enabled me to construct a patch. I want to mention that this ready and willing assistance is typical of their customer relations.

The Manual.

The manual is a very well written and comprehensive document which describes program operation. In fact, it is more like a treatise, first an easily understandable description of how to use the three programs, and then a thorough discussion of 6809 assembly language and program writing techniques; differences between 6800 and 6809 code; interfacing SDS80C with a printer; and use of the assembler with output from DISASM (Micro Works' disassembler for the Color Computer). They have done such a thorough job that they have made my writing this review in my own words very difficult. In brief, SDS80C is a well-thought out, well-implemented, and well-documented product.

Brickaway
by Jay Hoggins

DESCRIPTION Brickaway is similar to many games on the market for the Color Computer and other home computers and video games. A bouncing ball knocks "bricks" out of a wall at the top of the screen. With your joystick you control a paddle with which you attempt to keep the ball in motion. Brickaway gives you five balls. After they're gone the screen displays your score, ie the number of bricks you have knocked out.

When you start the game a yellow "wall" of "bricks" appears. Your ball begins moving and the game has begun. The object is to destroy the wall completely. When you've destroyed the yellow wall a blue one is drawn. After the blue one comes a red one. I've never gained enough skill to get the red wall destroyed. Maybe you get another color after that. I don't know if the program limits the number of walls and thus points or not. Maybe you could play indefinitly!

The ball seems to move at the same rate regardless of how many bricks you've knocked out. However, when the blue wall appears, it's lower edge is closer to your paddle than the lower edge of the yellow wall was. Likewise the lower edge of the red wall is closer than that of the blue wall. It takes less time for the ball to bounce from the wall back to the bottom and therefore you've got to be faster to hit it with the paddle. After you've eliminated the lower parts of the wall you've got more time so things seem to slow down.

Two more things need to be mentioned. First, in the instructions for the game it states, "when the ball is bouncing off the paddle vertically, and the paddle is at the left or right extreme of the screen, simply move joystick in the direction of the adjacent side, to "unstick" it." When I read this I didn't know what it meant. What happens is this: at either left or right ends of the screen the ball can take a straight vertical bounce. It knocks out a brick and comes back to the paddle. Again it goes straight up and hits a brick or goes through the wall if there are no bricks. This will continue indefinately until you move the joystick TOWARDS the adjacent wall of the screen. The paddle won't move, it's already against the side of the screen. Hey, this gives you a good chance for a well deserved break. The ball isn't going anywhere until you're ready! When you move the joystick the ball will take an odd bounce off the paddle and you're underway again.

Second, the paddle has a peculiar "drifting" characteristic. When you move the joystick the paddle seems to be connected to it by a rubber band. It takes some skill to master paddle movement.

CONCLUSIONS-Brickaway was designed to run on 4K or 16K machines. I was a little disappointed that it didn'tuse a higher resolution graphics mode. The ball, the paddle and the bricks are all quite large. Even the 4K Color Computer could support a higher resolution graphics mode.

As first I didn't like Brickaway because of the "drifty" paddle. My scores for the game were incredibly low. However, as I continued to play I began to get a feel for how to use the joystick to control the paddle. Then my scores began to look better. I would conclude that the "drifty" paddle adds an element of challenge to the game.

Overall, I've enjoyed playing Brickaway. The sound effects are quite good, the low resolution is disappointing but O.K., the "drifty" paddle is now a challenge, the "sticking" vertical bounce is peculiar but a welcome chance to rest. The price makes Brickaway an affordable addition to your software library.

COLOR COMPUTER SYSTEMS SOFTWARE

EDITOR/ASSEMBLER

The Micro Works Software Development System (SDS80C) is a complete 6809 editor, assembler and monitor package contained in one Color Computer program pack! Vastly superior to RAM-based assemblers/editors, the SDS80C is non-volatile, meaning that if your application program bombs, it can't destroy your editor/assembler. Plus it leaves almost all of 16K or 32K RAM free for **your** program. Since all three programs, editor, assembler and monitor are co-resident, we eliminate tedious program loading when going back and forth from editing to assembly and debugging!

The powerful screen-oriented Editor features finds, changes, moves, copys and much more. All keys have convenient auto repeat (typamatic), and since no line numbers are required, the full width of the screen may be used to generate well commented code.

The Assembler features **all** of the following: complete 6809 instruction set; complete 6800 set supported for cross-assembly; conditional assembly; local labels; assembly to cassette tape or to memory; listing to screen or printer; and mnemonic error codes instead of numbers.

The versatile ABUG monitor is a compact version of CBUG, tailored for debugging programs generated by the Assembler and Editor. It features examine/change of memory or registers, cassette load and save, breakpoints and more. **SDS80C Price: $89.95**

MODEM COMMUNICATIONS

Make your Color Computer an intelligent printing terminal with off-line storage! The Microtext module is just what you'll need for:

- —Talking to a timeshare system or information service
- —Printing out what is received as it is received
- —Saving received text to cassette tape
- —Re-displaying the received text even while on-line
- —Communications with other computers
- —Using your computer as a general-purpose 300-baud terminal
- —Downloading programs from other computers

The Microtext module is a program pack containing not only firmware but a second serial port so that both your printer and modem can be connected at the same time. Microtext can be configured for any serial printer that will work with the Color Computer, even if it requires line feeds! But even if you don't have a printer, you can keep a permanent copy of your data by storing to cassette tape. Also, any Radio Shack/Centronics-compatible parallel printer may be used by adding the Micro Works' PI80C parallel interface.

For those of you with special terminal applications, Microtext has selectable parity; it sends odd, even, mark or space. With mark parity (which is default) you can send to computers requiring either seven or eight bits. All 128 ASCII codes can be sent. Exchange programs with other Color Computer users! Basic programs may be downloaded from other computers or timesharing systems.

You'll find many uses for this versatile module! Available in ROMPACK, ready-to-use, for **$59.95.**

MACHINE LANGUAGE

MONITOR TAPE: A cassette tape which allows you to directly access memory, I/O and registers with a formatted hex display. Great for machine language programming, debugging and learning. It can also send/receive RS232 at up to 9600 baud, including host system download/upload. 19 commands in all. Relocatable and reentrant. **CBUG Tape Price: $29.95**

MONITOR ROM: The same program as above, supplied in 2716 EPROM. This allows you to use the entire RAM space. And you don't need to re-load the monitor each time you use it. The EPROM plugs into the Extended Basic ROM Socket or the Romless Pak I. **CBUG ROM Price: $39.95**

SOURCE GENERATOR: This package is a disassembler which runs on the color computer and generates your own source listing of the BASIC interpreter ROM. Also included is a documentation package which gives useful ROM entry points, complete memory map, I/O hardware details and more. A 16K system is required for the use of this cassette. **80C Disassembler Price: $49.95**

LEARN 6809!

6809 ASSEMBLY LANGUAGE PROGRAMMING, by Lance Leventhal, contains the most comprehensive reference material available for programming your Color Computer. **Price: $16.95**

HARDWARE

PARALLEL O!

USE A PARALLEL PRINTER with your Color Computer! Adaptor box plugs into the serial port and allows use of Centronics/Radio Shack-compatible printers with parallel interface. Assembled and tested. **PI80C Price: $69.96**

ROMLESS PAK I — is an empty program pack capable of holding two 2716 or 2732 EPROMS, allowing you up to 8K of program! The PC board inside comes with sockets installed, ready to go with the addition of your custom EPROMs. **Price: $24.95**

32K RAM!

MEMORY UPGRADE KITS: Consisting of 4116 200ns. integrated circuits, with instructions for installation. **4K-16K Kit Price: $39.95. 16K-32K Kit** (requires soldering experience) **Price: $39.95**

SPARE PARTS — SAMs, 6809Es, RAMs, PIAs. Call for prices.

GAMES

Pak Attack — Try your hand at this challenging game by Computerware, with fantastic graphics, sound and action! Cassette requires 16K. **Price: $24.95**

Star Blaster — Blast your way through an asteroid field in this action-packed Hi-Res graphics game! Available in ROMPACK; requires 16K. **Price: $39.95**

Berserk — Have fun zapping robots with this Hi-Res game by Mark Data Products. Cassette requires 16K. **Price: $24.95**

Adventure — *Black Sanctum* and *Calixto Island* by Mark Data Products. Each cassette requires 16K. **Price: $19.95** each.

THE MICRO WORKS

GOOD STUFF!

MasterCharge/Visa Accepted
California residents add 6% tax.

P.O. BOX 1110, DEL MAR, CA 92014 (714) 942-2400

TELEWRITER by Cogintec
Reviewed by Moe Schwartz
Mars Landing Power Project

When I first saw the ad for the "Telewriter" in the "Color Computer News", I said to myself, "24 line of 50 characters and true lowercase and all those word processing features for just $49.95, that's too good to be true." I needed the Word Processor for my reports on an APPORPIATE TECHNOLOGY Project, a rural Wind Turbine Generator, that I am doing for the Dept. of Energy, so I sent in my money and I took my chances.

After a wait of about two weeks I received an impressive mailing envelope and hurriedly opened it. It contained a cassette, a 27 page "Telewriter Tutorial", a 34 page "Telewriter Reference Manual, and a registration form. I placed the cassette on the recorder and opened the tutorial. The instructions were clear and I had no difficulty in getting started.

When the tape was loaded the Main menu appeared on the screen. The format was the familiar Basic Reversed Video and I thought "Uhoh! another ripoff". The Tutorial says to enter the first letter of the cammand you desire, so I entered "C" for Create a text file. The screen went blank white (I have a black and white monitor) and a black rectangle appeared in the upperleft corner. I entered a line of text and there before my eyes, lo and behold, just what the ad said would appear did appear.

24 lines of 51 characters in UPPER and real lowercase (not reverse video).

The "TELEWRITER" is a machine language program that uses a high resolution graphics screen to draw the text on the screen. There are no descenders and some of the characters are a little stiff but the text is quite readable, even on my small (9 inch) monitor. This is truly an ingenious program and it utilizes the full capabilities of the COLOR COMPUTER.

The documentation is excellent and complete and was obviously composed using the "TELEWRITER". There are over 70 commands for various functions. You have full cursor controls using the arrow keys, deletion is possible forward of and behind the cursor with the Break key. The program has a Wordwrap feature, and Full Print Formatting including Line Spacing, Left and Right Margins, Number of Characters/Line, Upper and Lower Page Margins, Page Numbering, Waiting at the end of a page, and Chain Printing of taped text files. All of the Print Format Parameters can be embedded in the text and can be changed during printing.

Also included are some special commands for the EPSON MX-80, such as, Font control (I have a R.S. LP VII, but I can change fonts with the embedded DIRECT (ASCII/CONTROL codes) Command, as well as Graphic Control codes, and a Baud rate control.

The MAIN MENU COMMANDS (one key) are:

C=Create a new text file. This destroys any old text in the buffer so the Program asks "SURE??" before execution.

E = Jump back to the Editor. Puts you back where you left off.

F = Jump to the Print Format Menu.

S = Save all text in the buffer to tape.

% = Save only the marked percentage of the text to tape.

R = Reads a text file from tape into the buffer (also asks "SURE??" before execution).

A = Appends a text file from tape to the end of the text buffer.

V = Verify or SKIPF to the end of the tape file, same as the SKIPF command in COLOR BASIC.

W = Word and Line count for all or part of the text.

Some of the EDITOR COMMANDS are:

A = Align text lines to fit the 51 character screen.

C = Copy a block of text.

F = Find a specified pattern.

G = Global (one at a time with one key repeat) search.

K = Kill a text line.

P = Page through the text.

R = Replace pattern.

X = Delete a block of text.

The "TELEWITER" makes writing a report, a review, or just a letter to your Mother easy and what's more LOTS of fun. After only a few hours of reading and playing with the "TELEWRITER" program I believe that it is one of the best programs for the Color Computer I have seen so far, and whoever is responsible for it deserves a lot of praise, as well as a lot of sales. If you do any writing at all, you should have this program.

TELEWRITER

Provides your COLOR COMPUTER with:

REAL LOWER CASE CHARACTERS ■ A POWERFUL TEXT FORMATTER
51 COLUMN × 24 LINE SCREEN DISPLAY ■ SPECIAL DRIVER FOR EPSON MX-80
ADVANCED CASSETTE HANDLING FEATURES ■ A SOPHISTICATED FULL-SCREEN TEXT EDITOR

and requires absolutely no hardware modifications

TELEWRITER

Telewriter is a powerful word processor designed specifically for the Color Computer. It can handle almost any serious writing job and it is extremely easy to use. It has all the advanced features you need to create, edit, store, format and print any kind of text. With Telewriter you can quickly produce perfect, finished copy for letters, reports, term papers, articles, technical documentation, stories, novels, screenplays, newsletters. It is also a flexible and efficient way to take notes or organize ideas and plans.

51 × 24 DISPLAY

The Color Computer is an incredibly powerful and versatile computer, but for text editing it has some major drawbacks. The small 32 character by 16 line screen format shows you too little of the text and, combined with its lack of lower case letters, bears little resemblance to the way text really looks on the page. Reverse video in place of lower case just adds confusion.

Telewriter eliminates these shortcomings with **no hardware modifications required**. By using software alone, Telewriter creates a new character set that has **real lower case letters**, and puts 24 lines of 51 characters on the screen. That's more on-screen characters than Apple II, Atari or TRS-80 Model III. That's more than double the Color Computer's standard display.

FULL SCREEN EDITOR

The Telewriter editor is designed for maximum ease of use. The commands are single key (or single key plus control key), fast, and easy to remember. There is no need to switch between insert modes and delete modes and cursor movement modes. You simply type. What you type is inserted into the text at the cursor, on the screen. What you see on the screen is always the current state of your text. You can move quickly through the text with one key cursor movement in all 4 directions, or press the shift key simultaneously for fast, auto-repeat. You can jump to the top or bottom of the text, the beginning or end of a line, move forward or backward a page at a time, or scroll quickly up or down. When you type past the end of a line, the wordwrap feature moves you cleanly to the next.

You can copy, move or delete any size block of text, search repeatedly for any pattern of characters, then instantly delete it or replace it with another. Telewriter gives you a tab key, tells you how much space you have left in memory, and warns you when the buffer is full.

FORMAT FEATURES

When it comes time to print out the finished manuscript, Telewriter lets you specify: left, right, top, and bottom margins; line spacing and lines per page. These parameters can be set before printing or they can be dynamically modified during printing with simple format codes in the text.

Telewriter will automatically number pages (if you want) and automatically center lines. It can chain print any number of text files from cassette without user intervention. You can tell it to start a new page anywhere in the text, pause at the bottom of the page, and set the Baud rate to any value (so you can run your printer at top speed).

You can print all or any part of the text buffer, abort the printing at any point, and there is a "Typewriter" feature which allows you to type straight to your printer. Because Telewriter lets you output numeric control codes directly (either from the menu or during printing), it works with any printer. There's even a special driver for the Epson MX-80 that lets you simply select any of its 12 fonts and do underlining with a single underline character.

CASSETTE HANDLER

Telewriter makes cassette as simple to use as possible. It will search in the forward direction til it finds the first valid file, so there's no need to keep retyping a load command when you are lost in your tape. You can save all or any part of the text buffer, and you can append pre-existing files to what you have in the buffer already. You can abort an append or filesearch without harming the program or the text in the buffer.

Telewriter will maintain compatibility with popular Color Computer disk systems, but, since it makes using cassette almost painless, you can still have a powerful word processor without the major additional cost of a disk.

AVAILABLE NOW

Telewriter turns your Color Computer into the lowest cost hi-power word processor in the world today. It runs in 16K or 32K (32K recommended) and is so simple you can be writing with it almost immediately. It comes with complete documentation and is fully supported by Cognitec. Telewriter costs $49.95 (California residents add 6% tax). To order or request more information write:

Cognitec
704 Nob Ave.
Del Mar, Ca. 92014

Or call (714) 755-1258 (weekdays, Saturdays, and early morning). We will gladly answer your questions.

PICTURE THIS
OR
PRINTER GRAPHICS MADE EASY

by Kathy Goebel

So you finally bought an LP VII. You've gleefully LLISTED all your programs and revised some of them to use the printer. You're so excited about having a printer that you're running all over the house proudly showing your latest listing to spouse/ parent/ friend/ dog- all of whom ought to be duly impressed. Somehow, even their inevitable lack of enthusiasm fails to daunt your euphoria. Well, how would you like to really knock their socks off? You can, with printer graphics.

The LP VII is a dot matrix graphics printer. Look closely at the way the printed letters are formed. They are made up of tiny dots. Through program statements, you can have control of each of these dots and create high-resolution pictures. And you can do it all in Basic.

The manual for the LP VII contains a very good introduction to using the graphics capabilities. However, it is geared to those who have a knowledge of the binary and hexadecimal number systems-a topic that can be confusing to the novice programmer. Furthermore, the manual does not include detailed examples which might better help one get started. In this article, I will present some tables, worksheets and examples designed to aid the novice as well as the more experienced programmer.

Before you can perform printer graphics with your Color Computer, you must have a printer driver, a short machine language program which is allegedly available free from Radio Shack (Cat. No. 70-2013). Once you CLOADM it and set the input select button on the back of your printer to 8BS, you're ready. Printing a CHR$(18) signals the printer to begin the graphics mode. You can return to text mode any time by printing a CHR$(30).

Each graphics print space is a column of 7 tiny dots, some of which may be "white", and therefore invisible on the paper and some of which may be black. You determine which dots are to be black and which "white" by calculating the numeric value of the column. This is done by adding 128 to the sum of the individual values of those dots which are to be black. These individual values are merely the powers of 2 as follows:

dot # value
1 2 0=1
2 2 1=2
3 2 2=4
4 2 3=8
5 2 4=16
6 2 5=32
7 2 6=64

Why add 128? There are actually 8 dots in the column. The eighth dot is always black for graphics, but it won't show up on the paper. It's value is 2 7, or 128.

There are 128 different ways to combine "white" and black dots in a 7-dot column. To spare you the drudgery of having to calculate the value for any given combination, I have devised a printer graphics reference table (Figure 1). This shows the graphic representation and corresponding value for each of the 128 combinations. The 0's represent "white" dots, and 2's, black dots. (See Listing 1 for the program which generates this table.)

Knowing which values we want, we can put them into our program like this

xxx PRINT#-2,CHR$(value);

Now that we know how to make combinations of dots, how do we put them together to form a picture? One way is to draw the picture on graph paper with each square representing a dot. Then divide the picture horizontally into sections of seven dots each. For each column of 7 dots, see which squares have a pencil mark (black dots) and which don't ("white" dots). Then use the Printer Graphics Reference Table to put the appropriate values in your program. Of course, odds are that you'll either go blind or crazy or both before you're finished.

A less frustrating method is to create the picture on the screen, either with SET/RESET or the Extended Basic commands LINE, DRAW, CIRCLE, etc. Then use a screen print routine which scans each area of the screen, calculates a graphic value and prints the picture. The following examples use this screen print method for both low and high resolution pictures.

Low Res

Figure 2 shows 8 blocks of varying shades, or "colors". The program (see Listing 2) creates these blocks in color on the screen with SET statements. The shades for the printed blocks simulate each of the colors on the screen and are purely arbitrary. The background color, 0, is black on the screen but "white" on the paper. Since no dots are to be black, the value for these columns is 128 (line 500). I chose color 1 (green on the screen) to be solid black on paper (all 7 dots are black). Therefore, each of these columns has a value of 255 (line 510). The remaining 7 shades (lines 520-580) are just sequences I thought might contrast nicely. I found that the best way to map out attractive shading patterns is to use a worksheet like the one in Figure 2a. (Listing 2a shows how to generate the worksheet.)

The size of the printed blocks is also arbitrary. Those in this program are four columns wide and seven dots high (the height must be a multiple of 7). Note that the screen print routing (lines 300-370) only scans a portion of the screen. Having mapped out the blocks on a SET/RESET worksheet, I knew that the picture would not extend beyond the 31st horizontal of the 23rd vertical positions. To scan the entire screen, change lines 320 and 330 to read:

320 for y=0 to 31
3310 for x=0 to 63

High Res

With Extended Basic, the possibilities are staggering. Draw or trace a picture on the 256X192 Graphics screen worksheet found in the back of "Going Ahead With Extended Color Basic." Using the graphics commands, create the picture on the screen (Figure 3). I found it convenient to plug the coordinates into DATA statements and use LINE commands to connect them (Listing 3). Don't be misled by the negative values. I only use them as a signal to start a new line ("Pen up"), rather than continue drawing from the previous point ("Pen down").

The screen print routine can then scan each of the 256X192 pixels. Each pixel represents a graphics dot. The program divides the screen horizontally into groups of 7 pixels, or dots, I then calculated the graphics value for each 7-dot column. Again, this example does not scan the whole screen-only the area containing this picture. To scan the whole screen, change line 160 to

160 for y=0 to 191 step 7

The number of ways printer graphics can be used is endless. Create a logo or design for your own letterhead. Chart you biorhythm. Draw up "blueprints" for your dream house. Map out a D&D adventure. Do landscapes, still life, portraits-the sky's the limit! Go for it, kiddo-you'll knock their socks off.

```
5 PRINT#-2,"PRINTER GRAPHICS R
  EFERENCE TABLE";TAB(65);"PAG
  E 1"
6 PRINT#-2:PRINT#-2
10 DIM B(6,16)
15 REM SET UP TABLE OF POWERS
   OF TWO
20 FOR I=0 TO 6
30 T(I)=INT(2^I)
40 NEXT I
50 FOR I=0 TO 127 STEP 16
60 FOR J=1 TO 16
70 L=I+J-1
80 FOR K=6 TO 0 STEP -1
90 IF L<T(K) THEN B(K,J)=0 ELS
   E B(K,J)=1:L=L-T(K)
100 NEXT K
110 NEXT J
120 FOR K=0 TO 6
130 FOR J=1 TO 16
140 PRINT#-2,USING"####";B(K,J
    );
150 NEXT J
160 PRINT#-2
170 NEXT K
175 PRINT#-2
180 FOR J=I TO I+15
190 PRINT#-2,USING"####";128+J
    ;
200 NEXT J
210 PRINT#-2:PRINT#-2:PRINT#-2
220 IF I<>48 THEN 300
230 FOR L=1 TO 19:PRINT#-2:NEX
    T
240 PRINT#-2,"PRINTER GRAPHICS
     REFERENCE TABLE";TAB(65);
    "PAGE 2"
250 PRINT#-2:PRINT#-2
300 NEXT I
```

```
10 PRINT#-2,"PRINTER GRAPHICS
   WORKSHEET"
20 PRINT#-2:PRINT#-2
30 FOR C=1 TO 3
40 PRINT#-2,"VALUE"
50 FOR I=0 TO 6
60 PRINT#-2,USING "###";INT(2^
   I);:PRINT#-2,TAB(6);
70 FOR J=1 TO 8
80 PRINT#-2,"     o";
90 NEXT J
100 PRINT#-2
110 NEXT I
120 FOR J=1 TO 8
130 PRINT#-2,TAB(6);CHR$(18);S
    TRING$(12,128);STRING$(18,
    129);
140 NEXT J
150 PRINT#-2,CHR$(30)
160 PRINT#-2," SUM="
170 PRINT#-2:PRINT#-2,TAB(6);
180 FOR J=1 TO 8
190 PRINT#-2," +128";
200 NEXT J
210 PRINT#-2
220 FOR J=1 TO 8
230 PRINT#-2,TAB(6);CHR$(18);S
    TRING$(12,128);STRING$(18,
    129);
240 NEXT J
250 PRINT#-2,CHR$(30)
260 PRINT#-2
270 PRINT#-2,"TOTAL"
280 PRINT#-2,"VALUE="
290 PRINT#-2:PRINT#-2
300 NEXT C
```

```
5 REM ENTER SHADE VALUES
10 FOR C=0 TO 8
15 C$(C)=""
20 FOR J=1 TO 4
30 READ A:C$(C)=C$(C)+CHR$(A)
40 NEXT J,C
50 REM DRAW PICTURE
60 CLS0
70 FOR Y=0 TO 3
80 FOR X=12 TO 19
85 SET (X,Y,1)
90 NEXT X,Y
95 FOR Y=4 TO 7:FOR X=16 TO 23
100 SET(X,Y,2)
105 NEXT X,Y
110 FOR Y=8 TO 11:FOR X=20 TO
    26
115 SET(X,Y,3)
120 NEXT X,Y
130 FOR Y=14 TO 17
135 FOR X=2 TO 9:SET(X,Y,4):NE
    XTX
140 FOR X=10 TO 17:SET(X,Y,5):
    NEXTX
150 FOR X=22 TO 29:SET(X,Y,6):
    NEXTX
160 NEXT Y
170 FOR Y=18 TO 21
180 FOR X=6 TO 13:SET(X,Y,7):N
    EXTX
190 FOR X=14 TO 21:SET(X,Y,8):
    NEXTX
200 NEXT Y
300 REM SCREEN PRINT ROUTINE
310 PRINT#-2,CHR$(18);
320 FOR Y=0 TO 31
330 FOR X=0 TO 63
340 PRINT#-2,C$(POINT(X,Y));
350 NEXT X
360 PRINT#-2
370 NEXT Y
500 DATA 128,128,128,128
510 DATA 255,255,255,255
520 DATA 145,196,145,196
530 DATA 145,128,196,128
540 DATA 213,170,213,170
550 DATA 201,146,164,201
560 DATA 179,230,204,153
570 DATA 145,136,196,162
580 DATA 199,156,241,199
```

Variable Name Lister
by C. J. Roslund

A Variable Lister is simply a utility that lists all variable names used by a given program. This is very useful when you want to make changes or a additions to a program of any length, or just want to document all variable names used in a program.

This utility will provide a listing of all SIMPLE (non-ARRAY) variables defined by your program (String and Numeric). Array variable names should be easy enough to locate from a DIMension statement near the beginning of the program.

There is one important requirement for the proper use of this utility; the BASIC program must have just been run and it must have addressed / defined all its variables at least once. (You may exit your program with a <BREAK>)

This requirement stems from the way the variable lister works. It scans the Basic variable storage space and prints the names of all variables found there.

The Variable Lister is a statement you must enter in the direct mode. It is only about 145 characters long (That's a pretty short utility program). It can list variable names to the screen or to a printer.

HERE IT IS

```
FOR I=PEEK(27)*256 + PEEK(28) TO
PEEK(29)*256 + PEEK(30)-5 STEP 7:
?#-2,CHR$(PEEK(I));: ?#-2,CHR$(PEEK(I+1) AND
127);: IF PEEK(I+1)>127 THEN ?#-2,"$": NEXT
ELSE ?#-2: NEXT
```

Remember you must run your program and get it to define all its variables before entering this statement. The above statement will list variable names to the printer. If you would rather have your variables listed to the screen just delete all the #-2s in the statement. This routine will work on Color Computers with or with Extended Basic. It may be entered without any spaces, although I inserted some for ease of reading.

I hope this utility proves to be as useful to you as it has been for me.

BUSTOUT

by Andrew Hubbell

I've decided it's time to do another critical review of Radio Shack's Program Paks only this time with a difference.

IN BUSTOUT Radio Shack has taken a popular video game and done a very poor job of implementation on the Color Computer. The graphics are pretty good, and the two player partnership, non-gravity game creates some interesting moments, but, for the most part, the game stinks.

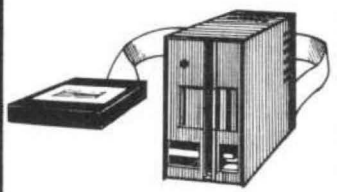
There are two basic flaws in this game: paddle movement and ball movement. Radio Shack, along with many other software authors, still has not figured out what to do with analog joysticks. The result is that paddle movement is very touchy and choppy. Two dimensional movement does nothing for the quality of the game, except allow you to knock the ball backward off the screen. Movement in the second dimension is used to alter the ball's speed, but the joysticks are so touchy that you have virtually no real control here. Ball movement also appears choppy. Once it contacts the rear wall it bounces almost straight back, almost too fast to see. At other times I have had the ball moving so slow, that I was able to put the joystick down, clean, fill and light my pipe between successive hits. I have also seen the ball moving at nearly a 90 degree angle to the field of play, rebound between side walls a dozen or more times between the paddle and the bricks, again a rather boring situation. Nor do I enjoy watching the ball mow down a double row of bricks when it happens to slip in between them.

Altogether, I feel the BUSTOUT, like most other program Paks I have seen, is worth only a small fraction of the $30 Radio Shack wants for it (fortunately, I did not spend my money on this one). As for the difference I mentioned above, I am demonstrating what I mean. Following this review you will find a listing of **BREAKAWAY**, a 4K BASIC version of the game that I wrote a few months ago. At the time I refused to market my version because the interpretive language caused movement to be too choppy. In many respect, my version is as good as what Radio Shack is peddling. I personally have too much pride to try selling merchandise of this quality, but I will give it away. So try it and judge for yourself. BUSTOUT is not much better.

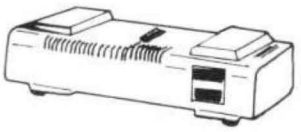
```
 0001 POKE65495,0
 0005 N=8:S=0:S0=1
 0010 S1=S0: S0=S0+0.5:
IFN<10THENN=N+1
 0015 CLS0: PRINT"BUSTDOWN     ";HI:
PRINT@20,S;: PRINT@29,N;: FORX=0TO31:
POKE1056+X,191: POKE1088+X,159:
POKE1120+X,143: POKE1152+X,175: NEXT:
B=128: GOSUB60
 0020 V=RND(6)+9: SP=S1: S1=S1+.4:
VM=1: HM=RND(0)+.5: H=0: Y=V: X=H
 0025 GOSUB60: IFPEEK(65280)AND1GOTO
25
 0030 GOSUB60: FORK=1TOSP: RESET(X,Y)
 0032 C=H+HM: R=V+VM: X=INT(C):
Y=INT(R): IFY>31GOTO50
 0034 IFX<00RX>63THENHM=-HM:
SOUND120,1:GOTO32
 0036 IFY<30GOTO40
ELSEL=1504+INT(X/2): IFL<P
ORL>P+4GOTO40
 0038 IFL=P ANDHM>00RL=P+4
ANDHM<0THENHM=-HM ELSEIFL=P
ANDHM<00RL=P+4 ANDHM>0THENHM=HM*1.7
ELSEIFL=P+1 ANDHM>00RL=P+3
ANDHM<0THENHM=-HM*0.7
 0039 VM=-VM: SOUND60,1:GOTO32
 0040 IFY>9GOTO48
ELSEIFPOINT(X,Y)=0GOTO48
 0042 A=1:
IFY>1ANDY<8THENA=8-2*INT(Y/2)
 0044 S=S+A: PRINT@20,S;: SOUND200,1:
IFY>1 THENL=1024+32*INT(Y/2)+
INT(X/2): POKEL,192: B=B-1: IFB<1THEN
FORI=0TO9: SOUND200,2: NEXT: GOTO10
 0045 IFVM<0THENVM=-VM: SP=SP+0.15
 0046 GOTO32
 0048 SET(X,Y,5): H=C: V=R: NEXTK:
GOTO30
 0050 N=N-1: PRINT@29,N;: IFN>0
GOTO20
 0052 IFS>HI THENI=HI: HI=S:
IFI>0THEN PRINT@320,"     NEW HIGH
SCORE  ";S
 0054 GOSUB60: R$=INKEY$:
IFR$=""GOTO54 ELSEIFASC(R$)=13
GOTO5ELSE54
 0060 I= INT(JOYSTK(0)*.435+.5)+1504:
IFI=P THENRETURN ELSEFORJ=0TO4:
POKEP+J,224: NEXT: P=I: FORJ=0TO4:
POKEP+J,236: NEXT: RETURN
```

From Computer Plus to YOU . . .

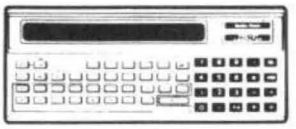
PLUS *after* PLUS *after* PLUS



Color Computer Disc Drive
$470



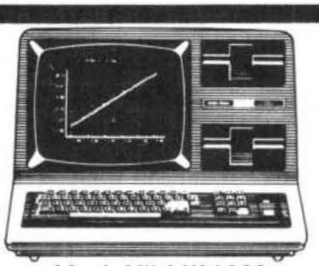
32K Exp. Int. $334



Pocket Computer $189
P.C. Printer $130

Color Computer 4K $310
w/16K Ext. Basic $459

MODEL II 64K $3300



Model III 16K $839
Model III 48K
2 Disc & RS232C $2100

BUY DIRECT *Here are just a few of our fine offers . . . call TOLL FREE for full information.*

COMPUTERS	
Model II 64K	$3300
Model III 4K LEV I	599
MODEL III 16K	839
MODEL III 32K	945.50
*MODEL III 32K	881.50
MODEL III 48K	1052
*MODEL III 48K	924
Model III 48K 2 Disc & RS232 c	2100
Color Computer 4K	310
Color Computer 16K	416.50
*Color Computer 16K	352.50
Color Computer 16K w/extended basic	459
Color Computer Drive 0	470
Pocket Computer	189
VIDEOTEX	310

*Computer Plus New Equipment. 180 Day Extended Warranty

PERIPHERALS	
Expansion Interface OK	$249
Expansion Interface 16K	355.50
*Expansion Interface 16K	291.50
Expansion Interface 32K	462
*Expansion Interface 32K	334
16K RAM N.E.C. 200 N.S. chips	25
MODEMS	
Lynx Direct Connect MI/MIII Auto Ans./Dial	269
Telephone Interface II	169
R.S. Modem I D.C.	130
R.S. Modem II D.C.	199
PRINTERS	
Daisy Wheel II	1695
Line Printer VI	999
Epson MX80	499
Epson MX80 FT	599
Epson MX100	799
Line Printer VII	315
Line Printer VIII	620
Line Printer V	1610
Microline 80	345
Microline 82A	515
Microline 83A	799
Pocket Computer Printer	130
DISK DRIVES	
R.S. Model III 1ST-Drive	712
TEAC 40 Track MI	329
R.S. 1 Drive Exp MII	999
R.S. 2 Drive Exp MII	1518
R.S. 3 Drive Exp MII	2040
SOFTWARE	
R.S. Software 10% off list	
Newdos + (40) track MI	88
Newdos 80 MIII	149
ST80III	149
ETC.	
Verbatum 5" Double Density	32
Verbatum 8" Data Life	49.95
Ctr-80A recorder	52
C. C Joysticks	22

We have the lowest possible Fully Warranteed Prices AND a full complement of Radio Shack Software.

DEALER INQUIRIES ARE INVITED

Prices subject to change without notice.
Not responsible for typographical errors.

TRS-80 is a registered trademark of Tandy Corp.

call TOLL FREE 1-800-343-8124

computer plus

245A Great Road
Littleton, MA 01460
617-486-3193

Write for your free catalog

CO-RES9 is a cassette based Editor Assembler for the Color Computer requiring at least 16K of RAM. It's my humble opinion that an assembler is an assembler and the real crux of writing assembly language programs is in the editor and this is where CO-RES9 really struts its stuff.

In my experience CO-RES9 is the only Editor on the market now that will allow you to edit Basic programs in addition to Assembly language. Before we get into that lets look at the editing commands.

LIST is essentially the same as Basic's LIST command in that it will either list the entire program or will list a specified range of lines.

NLINES removes all line numbers for the sake of listing or printing the program file.

TLOAD is the cassette load command. TLOAD will load either its own files or BASIC files saved in ASCII format.

TSAVE is the opposite of TLOAD.

TAPPEND is a merge type command that allows the appending of two or more files.

RESEQUENCE is a renumber utility that allows you to cleanup your code prior to anyone looking it over. Any true AL programmer will agree that sloppy looking code will run but usually you want to tidy it up a bit before you show it to the world.

DELETE allows the removal of blocks of code or lines of program file. The command is identical to BASIC's DEL..

SEARCH looks for all occurances of a string within specified line numbers and reports them so that you may edit the lines or whatever.

RPLACE will search and replace all occurances of a specified string with a new string within specified line numbers this is handy for changing all occurances of a particular label or for changing all of the same type spelling errors at the same time.

LEDIT is the command for editing a line of code.

AEDIT is an automatic editing function that starts at a specified line number and continues editing lines until you force it to stop. This is great of making massive changes in the program file.

COPY allows you to repeat certain lines without having to retype them sort of a primitive macro type function.

MOVE will take lines from one area of a program to another area, if you go back to my comment about tidying up code this is the real workhorse of making presentable code. Many times you will realize that a listing will make much more sense if a block of code or a subroutine were move and this is the function to make the move fast and easy.

SIZE shows the amount of memory currently in use (the opposite of PRINT MEM).

PRINTER forces the next command to use the printer or RS-232 port.

NEW deletes all of the file currently in memory.

AUTO provides an auto incrementing line numbers (the feature RS BASIC forgot). AUTO is especially useful for those times that you're really hot and laying down code like mad..

RLINES removes the line number from the file currently in memory. I think this command is best used before printing the listing, personal opinion, AL programs look much better without line numbers

ILINES inserts line numbers into the program file in memory. Handy for putting them back after making the printed listing.

SKIP is the same as BASIC's SKIPF function in that it will skip over cassette file until it passes the specified file.

PATCH causes an SWI3 to be executed, effectively causing a jump to your monitor.

RUN allows testing object code in memory.

All of the functions work extremely well and I have as yet been unable to find a bug in the program.

In addition to being an excellent Editor Assembler program it works well as a BASIC editor program. The RPLACE function works extremely well for things like correcting spelling in a BASIC program (there are some words that I consistantly spell wrong) and for merging common subroutines into programs.

All in all I find CO-RES9 to be an excellent Assembler program with many editor features now found in some of the more expensive Assemblers on the market, in fact I hesitated in purchasing CO-RES9 for some time based on the price. Occasionally you do get more than what you pay for!

Othello

by Tony Di Stefano

```
10 CLS(0)
20 FOR I=5 TO 58: SET(I,0,5):
   SET(I,26,5): NEXT I
30 FOR J= 1 TO 25: SET(5,J,5):
    SET(6,J,5): SET(57,J,5): S
   ET(58,J,5): NEXT J
40 FOR J= 0 TO 1: FOR B= 2 TO
   23 STEP 3: FOR I=0 TO 3: FO
   R A=9 TO 51 STEP6
50 SET(A+I,B+J,1)
60 NEXT A,I,B,J
65 FORI=1472 TO 1535:POKEI,96:
    NEXT I
70 DIM A(8,8)
75 W=60
80 FOR X=0 TO 7: FOR Y=0 TO 7:
   A(X,Y)=1: NEXT Y,X
90 A(3,3)=4: A(4,4)=4: A(4,3)=
   3: A(3,4)=3
100 X=4: Y=3: GOSUB1000
101 X=3: Y=4: GOSUB1000
102 X=3: Y=3: GOSUB1000
103 X=4: Y=4: GOSUB1000
120 GOSUB 2000
200 PRINT@448,"
                        ";
201 PRINT@448,"YOUR MOVE - RED
     "; : CR=0 : SOUND 150,5
210 Y=INT(JOYSTK(0)/8.1) : X=I
    NT(JOYSTK(1)/8.1)
215 RESET(Y*6+9,X*3+2)
216 M$=INKEY$
217 IF M$=" " THEN GOSUB 5000
220 FOR I = 1 TO 30 : NEXT I
225 SET(Y*6+9,X*3+2,A(X,Y))
230 IF PEEK(65280) = 126 OR PE
    EK(65280) = 254 THEN 250
240 GOTO 210
250 PRINT@480,"
                        ";
260 XX=X : YY=Y
270 C=4 : IF A(X,Y) = 1 THEN 5
    00
280 GOTO 200
300 PRINT@448,"
                        ";
301 PRINT@448,"YOUR MOVE - BLU
    E"; : CR=0 : SOUND 150,5
305 Z=JOYSTK(0)
310 Y =INT(JOYSTK(2)/8.1) : X=
    INT(JOYSTK(3)/8.1)
315 RESET(Y*6+9,X*3+2)
316 M$=INKEY$
317 IF M$=" " THEN GOSUB 5000
320 FOR I = 1 TO 30 :NEXT I
325 SET(Y*6+9,X*3+2,A(X,Y))
330 IF PEEK(65280) = 125 OR PE
    EK(65280) = 253 THEN 350
340 GOTO 305
350 PRINT@480,"
                        ";
360 XX=X : YY=Y
370 C=3 : IF A(X,Y) = 1 THEN 5
    00
380 GOTO 300
400 IF C=3 THEN C=4: GOTO 450
420 IF C=4 THEN C=3: GOTO 450
450 REM
470 GOTO 2500
500 REM
510 FOR DR=0 TO 7: IC=1
515 GOSUB 1100
525 IF TC=-1 OR TC=1 OR TC=C T
    HEN GOTO 590
530 FOR IC=2 TO 7: GOSUB 1100
535 IF TC=-1 OR TC=1 THEN GOTO
     590
545 IF TC=C THEN GOSUB 1200: G
    OTO 590
580 NEXT IC
590 NEXT DR
595 IF CR=0 THEN GOTO 810
800 IF CR=1 THEN GOTO 400
810 IF C=3 THEN A$=" BLUE!": G
    OTO 830
820 A$=" RED"
830 PRINT@480, "YOU ARE NOT OU
    TFLANKING";A$;
835 SOUND55,14
840 IF A$= " RED" THEN GOTO 20
    0
850 GOTO 300
1000 FOR J=0 TO 1: FOR I=0 TO
     3
1020 SET((Y*6+9)+I,(3*X+2)+J,A
     (X,Y))
1025 NEXTI,J
1030 RETURN
1100 IF DR=0 THEN Q=1: P=0
1102 IF DR=1 THEN Q=1: P=1
1103 IF DR=2 THEN Q=0: P=1
1104 IF DR=3 THEN Q=-1:P=1
1105 IF DR=4 THEN Q=-1:P=0
1106 IF DR=5 THEN Q=-1:P=-1
1107 IF DR=6 THEN Q=0: P=-1
1108 IF DR=7 THEN Q=1: P=-1
1110 PX=XX: PY=YY
1115 FOR K=1 TO IC: PX=PX+Q: P
     Y=PY+P: NEXT K
1120 IF PX>7 OR PX<0 OR PY>7 O
     R PY<0 THEN TC=-1: GOTO 1
     140
1130 TC=A(PX,PY)
1140 RETURN
```

```
1200 FOR Z=1 TO IC
1220 A(X,Y)=C: POKE15115+Y+(X*
     10),C: GOSUB 1000: X=X+Q:
      Y=Y+P
1225 SOUND50+Z*21,1
1227 NEXT Z
1230 GOSUB 1000
1240 X=XX: Y=YY
1245 CR=1
1250 RETURN
1305 CLS
1320 PRINT:PRINT"           END
      OF GAME"
1330 FOR X=0 TO 7: FOR Y=0 TO
     7
1340 IF A(X,Y)=3 THEN BL=BL+1
     : GOTO 1360
1350 IF A(X,Y)=4 THEN RE=RE+1
     : GOTO 1360
1360 NEXT Y,X
1365 IF RE=BL THEN PRINT:PRINT
     "ITS A TIE!!!": GOTO 1410
1370 IF BL>RE THEN PRINT:PRINT
     "YOU WIN BLUE!!!"; : GOTO
      1390
1380 PRINT:PRINT"  YOU WIN RED
     !!!";
1390 TE=ABS(BL-RE)
1400 PRINT" BY"; TE
1410 PRINT:PRINT:PRINT:PRINT"
      BYE FOR NOW"
1420 SOUND RND(250),RND(6)
1430 GOTO 1420
2000 AD=15115
2010 FOR X=0 TO 7:FOR Y=0 TO 7
2020 POKE AD,A(X,Y)
2030 AD=AD+1
2040 NEXT Y
2050 AD=AD+2
2060 NEXT X
2070 FOR I=15104 TO 15113 :POK
     EI,0:POKEI+90,0 :NEXT I
2080 FOR I=15114 TO 15194 STEP
      10:POKEI,0:POKEI+9,0 :NE
     XT I
2100 FOR I=15344 TO 15352
2110 READ DT :POKEI,DT
2120 NEXT I
2130 FOR I=15360 TO 15471
2140 READ ML :POKE I,ML
2150 NEXT I
2200 RETURN
2500 POKE 15358,C
2510 EXEC 15360
2520 CT=PEEK(15357)
2530 IF CT=0 THEN 2600
2540 EN=0 : IFC=4 THEN 200
2550 GOTO 300
2600 IF EN=1 THEN 2700
2610 IF C=4 THEN PRINT@448,"NO
      MOVE POSSIBLE FOR RED";:
      SOUND 1,40: GOTO 2630
2620 PRINT@448,"NO MOVE POSSIB
     LE FOR BLUE";
2625 SOUND 1,40
2630 EN=1
2640 IF C=3 THEN C=4 :GOTO 250
     0
2650 IF C=4 THEN C=3 :GOTO 250
     0
2700 PRINT@480,"           END
     OF GAME";
2705 SOUND 100,20:SOUND10,20
2710 GOTO 1305
3500 DATA 1,11,10,9,255,245,24
     6,247,0
3510 DATA 246,33,240,79,183,59
     ,253,134,8,183
3520 DATA 59,252,134,88,183,59
     ,251,134,59,183
3530 DATA 59,249,182,59,251,18
     3,59,250,190,59
3540 DATA 249,166,132,128,1,38
     ,65,16,142,59
3550 DATA 240,166,164,187,59,2
     50,183,59,250,166
3560 DATA 159,59,249,129,0,39,
     23,129,1,39
3570 DATA 19,177,59,254,39,7,1
     34,1,183,59
3580 DATA 253,32,224,182,59,25
     3,129,1,39,31
3590 DATA 79,183,59,253,182,59
     ,251,183,59,250,166,160,1
     22,59
3600 DATA 252,38,200,134,8,183
     ,59,252,122,59
3610 DATA 251,38,171,79,183,59
     ,253,57
```

ONE, TWO, TREE

by Mr. Lynn Davis

I was really intriqued by Thomas Mix's proqram which produces a tree (Color Computer News, Nov/Dec 1981).

As I sat in my chair watchinq the "tree" qrow, I beqan wonderinq how many times a liqht would flash (a leaf?) before one actually became a part of the tree. It also seemed loqical to me that as the tree qrew, there would be a qreater likelihood of a "leaf" hittinq and stayinq on the tree. But how does one prove this?

The easiest way, of course, it to let the computer do the math. If you type in the proqram, or make the necessary modifications if you saved the proqram on tape, you will see a runninq total of the number of leaves on the tree, a runninq tally of the number of times the computer qenerates a leaf, and the percentaqe of leaves on the tree per computer attempts. A tone will also be qenerated to let you know when a leaf has been placed on the tree.

The first modification necessary is the addition of times a leaf is put anywhere on the screen and prints that total at location number 408.

Third, line number 33 computes the percentaqe of leaves that stick to the tree per number of attempts, and then prints that number at position 440. A PRINT USING statement was used so all the diqits are not printed. For those readers without extended basic, see the REMark in line 34. This PRINT command that people without extended basic can use just moves the printed percentaqe to the left so that all siqnificant diqits can be printed.

Fourth, lines 60, 70, 80 and 85 must be extended in order to qenerate a tone when a leaf has indeed been counted, and a PRINT @ command to show on the screen the total number of leaves stuck to the tree.

I have used Mr. MIx's oriqinal line numbers so that anyone who saved the proqram on tape can make the modifications without confusinq line number chanqes. In the proqram's final version you may want to renumber it so that you can add a CLS command in the beqinninq.

What's nice about the proqram now is that not only do you have a qreat little qraphics display, but you also have a vehicle by which kids can learn (or review) a little math. With this proqram you will visually see the relationship between numbers and the percentaqes they qenerate.

Usinq the SHIFT and @ key at the same time, you can stop the proqram. Kids can then calculate the percentaqes with paper and pencil and then "check the computer". If you have a PRINT USING command, this can also lead to a discussion of "roundinq off".

On a more advanced level, this proqram could also lead to a discussion on probability (i.e. why the percentaqe of leaves in the beqinninq is low, why the percentaqe increases as the size of the tree increases, and why the percentaqe qoes down once the tree is nearinq completion).

It's a neat little proqram that can brinq you, your children and the computer toqether for a little fun--and a little learninq.

```
1 PRINT @33,"THIS PROGRAM WILL BUILD A
TREE": PRINT@167,"BY THOMAS L. MIX": PRINT
@199,"3424 COLLEGE N.E.": PRINT @231,"GRAND
RAPIDS, MICH. 49595"
2 FOR R=1 TO 460*4
3 NEXT R
5 CLS(5)
6 D=1
10 CLS(0): FOR H=0 TO 63: SET(H,29,5): NEXT H
15 FOR V=10 TO 29: SET(31,V,5): NEXT V
18 H=RND(29): H=H+34
20 FOR V=15 TO 35 STEP -1: SET(31,V,5)
22 NEXT V
30 H=RND(62): V=RND(20)
32 C=1: C1=C+C1: PRINT@408,C1
33 PRINT@440,USING "##.##%"; (D1/C1)*100
34 REM WITHOUT EXTENDED BASIC CHANGE
LINE 33 TO: 33 PRINT @ 435,(D1/C1)*100
40 IF POINT(H,V) THEN 30
50 Q=RND(8): SET(H,V,Q)
60 IF POINT(H,V+1)THEN SOUND 50,1: D1=D+D1:
PRINT @376,D1;: GOTO30
70 IF POINT(H,V-1)THEN SOUND 100,1:
D1=D+D1: PRINT@376,D1;: GOTO30
80 IF POINT(H+1,V)THEN SOUND 150,1:
D1=D+D1: PRINT@376,D1;: GOTO30
85 IF POINT(H-1,V)THEN SOUND 200,1:
D1=D+D1: PRINT@376,D1;: GOTO30
90 RESET(H,V)
99 GOTO 30
100 REM
110 REM *************************
120 REM MODIFIED FOR MATH DISPLAY PLUS
SOUND
130 REM BY MR. LYNN DAVIS
140 REM  4315 AMBLEWOOD LANE
150 REM  CLAY, N.Y. 13041
160 REM *************************
```

Rotating Shapes
by John Heusinkveld

One of the most exciting claims in Radio Shack's advance descriptions of Extended Color Basic was that shapes could be rotated. Unfortunately, this proved to be a misunderstanding. The only way to rotate a shape is to create eight strings, one for each horizontal, vertical, and diagonal position and DRAW them one after another. When you have anything but the simplest of shapes, this is impossibly complicated.

I decided that what I needed was a subroutine to take a shape-defining string and change it so that the shape it defines would be rotated forty-five degrees in either direction. After a little work, I had a Basic routine to do just that, but it took several seconds to rotate a string, so it wasn't much use in real time applications.

Short of translating this routine into machine language, there is one solution to the speed dilemma; at the beginning of the program, read all of the possible rotations (there are eight for any figure) into a string array and then use a variable to point to the array element currently in use. To turn, add or subtract one from the current value of the pointer variable and redraw the string. If this has you hopelessly confused, enter and run program listing 1. This is an interesting demonstration program. It asks you to enter the speed at which your tank is to move and then seems to hang up for a half-minute or so. What it is doing in this time is creating all eight strings needed to display the eight possible tank directions. It also sets up two numeric arrays: XI and YI. XI stores the horizontal increment to move in any direction, and YI stores the vertical increments. When all of those arrays are ready, the high res graphics are switched on and it starts drawing the tank. To simulate the noise of engines, there is short PLAY command embedded in the loop. At this point, the player can turn in either direction using the corresponding arrows, or he may move forward by pressing the space bar. Since this is just a demonstration, there is no protection against leaving the screen, and, as there is nothing to shoot at, there is no provision for firing. Both of these could be added easily; in fact, without too much difficulty it could be expanded into a full-blown tank battle game, but I leave that to your ingenuity.

```
0 PCLEAR4:PMODE4,1:PCLS:CLEAR100
0:CLS:PRINT:PRINTTAB(10)"TANK DE
O"
10 Q$="UERFDGLH"
20 L$="U10D5R3D10L6U10R3":D=1:IN
PUT"SPEED";P
30 FORO=1TO8:T$(O)=L$:GOSUB1000:
NEXTO
40 YI(8)=-P:YI(1)=-P:YI(2)=-P:YI
(4)=P:YI(5)=P:YI(6)=P
50 XI(2)=P:XI(3)=P:XI(4)=P:XI(6)
=-P:XI(7)=-P:XI(8)=-P
60 PLAY"T255L25501V20"
70 SCREEN1,1:DR=1:TX=128:TY=96:
GOTO100
80 DRAW"C0"+PS$+T$(DR)
90 IFPEEK(345)=247THENTX=TX+XI(D
R):TY=TY+YI(DR)
100 A$=INKEY$:IFA$=CHR$(8)THENDR
=DR-1:IFDR<1THENDR=8
110 IFA$=CHR$(9)THENDR=DR+1:IFDR
=9THENDR=1
120 PS$="BM"+STR$(TX)+","+STR$(T
Y):DRAW"C1"+PS$+T$(DR)
130 PLAY"CC":GOTO80
140 END
1000 FORA=1TOLEN(L$):W$=MID$(L$,
A,1):FORC=1TO8:IFW$=MID$(Q$,C,1)
HEN1010ELSENEXTC:GOTO1040
1010 IFD=1THENC=C+1ELSEC=C-1
1020 IFC=0THENC=8ELSEIFC=9THENC=1
1030 MID$(L$,A,1)=MID$(Q$,C,1)
1040 NEXTA
1050 RETURN
```

$50 CC Modem

by Mark Lockwood N8AVB
3941 Boeing Dr.
Saginaw, MI 48604

After seeing an inquiry in one of the computer publications (ref.#1), as well as reading the original article from a year earlier, it was only natural that I, as an amateur radio operator with a new color computer, would want to explore another means of communications.

With a short phone call I found out that the kit from MicroMint (ref.#2) was an improved version on the article and still selling strong. During the two week wait for the kit to arrive I decided to build the Modem using an old telephone and isolation transformer instead of acoustic-coupler with Microphone and speaker.

When the kit arrived I found it to be well documented with approximately 12 pages of instructions, parts list, and schematics. Three evenings with a soldering iron, the kit and it's recommended power supply was ready to hook up to a telephone. At this point, I enlisted the help of WD8OQH Jerry, a good friend of mine, to whom I owe many thanks for much help on a number of radio and computer projects I have built in the past. One late night session of hunting for jumper leads, learning what makes a telephone work, and about 20 minutes of testing and calibration, Jerry and I were calling computers all over the State and running up Jerry's phone bill. The Modem worked flawlessly!!! Three more evenings mounting a fuse, switches, and cutting holes in a 3x7x12" chassis box, paint, labels, and final assembly set the completed Modem along side of C.C..

As to the actual construction: The layout appears not to be critical. I used a 1/4" thick piece of fiber board 7x9 1/2", on to which I mounted the completed Modem kit, the power supply, and isolation transformer. The board was then mounted to the open side of the chassis box thus forming a bottom with the components up inside the box. Putting 4 1/2" rubber feet on the corners of the box lips allows air to circulate through the box. No changes were made to the Modem Kit. However, I did use a 24 volt 300 Ma. transformer when building the recommended power supply (thats all my junk box had). At the speaker connections of the Modem Kit I used a 8 ohm to 1K ohm transformer (Radio Shack 273-1280) and at the microphone connections a 600 ohm to 600 ohm transformer (again from the junk box). Both connect to the telephone. I chose to mount the telephone to the box with 4 screws through the phone's feet and all wires to or from the phone go through a 1/4" hole under the phone into the box. On the front end of the box I mounted a power supply on/off switch, a small 120V power "on" lamp, the carrier detect LED. and a phone/modem switch. The back end has a 1/2 amp. fuse and holder, the line cord, RS-232 cable, and the telephone cord.

Use of the system is quite simple. Load and start Radio Shacks 26-2222 VideoTex program, pick up the telephone receiver and "place the call" at the prompt, and when data is received the LED lights, set the receiver down some where on the table, and press C.C.'s "enter" key.

Two final comments are: (1) get with a person familiar with your type of telephone for making the wire connections, there maybe many versions I am not aware of and (2) watch your telephone bill, it may jump, if you make many long distance calls. Time flies when your having fun with C.C..

Ref. #1 "Byte" August 1981 Pg. 388 and "Byte" August 1980 Pg. 22

Ref. #2 The MicroMint Inc., 917 Midway, Woodmere, N.Y. 11598, (516) 374-6793
Modem Kit of integrated circuits, all components, P.C. Board, and all Directions, $39.95 plus $2.00 shipping.

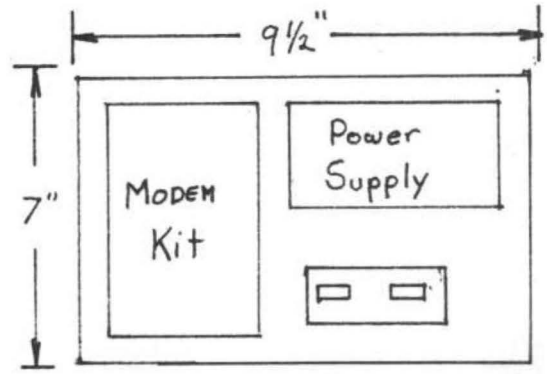
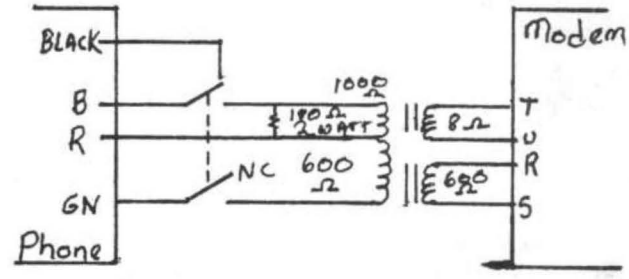


Figure #1



A TALE OF TWO BASICS

by D. S. Lewandowski

The Radio Shack Color Computer may be the answer to the computer hobbist's dreams. As everyone knows upgrading the ram to 32K is very easy, here is another simple modification.

Having extended basic is at times a disadvantage because of all the memory cleared for the graphic screens. Also some people may like to know if their programs would run under both extended and standard basic. With the addition of two wires and a small switch you can have both basic's in one machine.

The extended basic ROM in the Color Computer is U28. By adding a switch on pin 24 (the +5 vdc line) the ROM can be disabled with no ill effect on the machine. This also gives you the capability of running standard basic programs without any of the extended basic features to see if the program is compatible with both basic's.

To do the modification is simple if you are used to working with IC's and soldering. If not find someone who is and let them do it for you.

First remove the seven screws from the bottom of the Color Computer (NOTE: This will void you Radio Shack Warranty). Turn the unit upright and remove the top cover. Now cut the two (or three) wire ties which hold the rf shield in place. (Wire ties are white plastic straps) Carefully lift the top off the rf shield, it's large so do it slowly as not to bend it. Now locate U28 (This is the extended basic ROM) it's just in front of the video modulator (The silver box the TV cable plugs into). It has the marking U28 between the chip and the rf shield. CAREFULLY remove the chip from the socket. If you bend any pins you can reform them with needlenose pliars. Locate pin 24 on the chip and CAREFULLY bend it back over the top of the chip (see diagram 1), now reinsert the chip into the socket (U28).

Now get the switch (R/S #275-324) and twelve inches of two wire cable (R/S #278-755), solder one wire to each connector on the switch. (You may also wish to drill a hole for the switch in the top cabinet and mount the switch at this time.)

Feed the other end of the cable through the side of the rf shield near the program pak slot. Carefully insert one wire into pin 24 of socket U28 it should go into the socket about 1/8 ". Solder the other wire to pin 24 that is bent on top of U28. Place a peice of electrical tape over the connection when done.

Replace the rf shield, and install the switch (Mine is mounted two inches left of the reset button). Carefully replace the top cover (Remember the cable is only twelve inches long). Replace the seven screws the shortest two go into the holes opposite the space bar.

When you power up the machine, you should get either the Exteded basic logo you're used to, or just Color basic logo depending on the switch position. If you get anything else e.g. garbage .. Turn it OFF and recheck your work. Now turn the machine off and move the switch to the other position you should get the opposite logo of the first time.

PLEASE NOTE: No matter how simple a modification is you are still working with a delicate machine DO NOT attempt to do this modification IF you'll never use it, OR if you cannot solder, OR IF YOU CAN'T READ. I have seen good computers killed by owners using 350 WATT guns to solder IC chips.

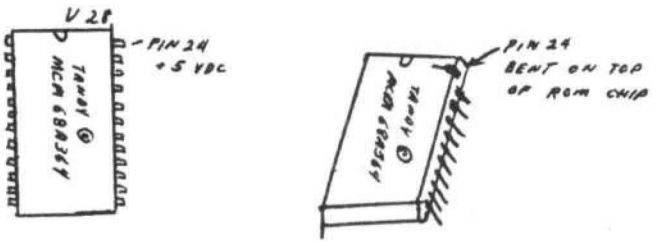


FIG. 1

80-U.S.

THE TRS·80 USERS JOURNAL

80-U.S. Journal is a monthly publication for the TRS-80 computer owner. The Journal covers Business, Scientific, Educational, and Recreational areas.

80-U.S. will keep you up to date on new products, software and hardware. Each issue will have listings of programs, reviews, tutorials. 80-U.S. is the complete "How to" Journal for the TRS-80!

If you haven't taken a look at 80-U.S., here is a no-risk opportunity to do it now. Become a trial subscriber now under the protection of a *full* money-back guarantee!

PLEASE enroll me as a trial subscriber to 80-U.S. and bill me just $16. I understand I reserve the right to cancel my subscription any time, for any reason, and receive a refund for the balance of my subscription. If I should decide to cancel after receiving my first issue I will simply mark my bill *cancel* and keep the first issue *FREE!*

Name _____

Address _____

City _____ State _____ Zip _____

☐ I prefer to enclose payment now.

Visa/MC # _____

Exp. Date _____

Canada & Mexico: $25 per year, other foreign subscriptions $30 surface mail, $72 per year via airmail.

80-U.S. Journal
3838 South Warner Street
Tacoma, Washington 98409
(206) 475-2219

Order # CCN

TRS-80 is a Registered Trademark of the Tandy Corp.

Interrupts

by Kenneth Kalish
85 Cooper St
Pringle, PA 18704

There are some occasions when you need to measure "real" time. For example, you might want to time how long it takes for a person to answer a series of questions in an education program, or you might use a clock as part of a game. A clock interrupt system allows you to do just that. With it you can build a software clock in RAM, which can then be used by the main program to time whatever you wish.

Simply stated, a clock interrupt works as follows. While the CPU is coursing along through normal program execution, it suddenly receives an interrupt signal. The CPU stops whatever it's doing, saves all of the registers (including the Program Counter) by pushing them on the stack, and jumps off to a special interrupt service routine. The service routine increments one or more clock counters, and finishes up by restoring all of the registers. Since the PC is also restored to its original value, control jumps back to the exact point in the main program where it had left off when the interrupt first occured.

The interrupt system is "wired" into hardware. The triggering pulse comes from a real time clock, and the stacking of registers as well as the jumping off to the service routine occurs automatically. But the system is also under software control. You can turn it off or on whenever you wish; and since the service routine is just a short block of program that you write yourself, it can do anything that you want it to do.

In order to use to interrupts on the Color Computer, it's important to understand more about how the interrupts and the PIAs work. There are three interrupts on the 6809; the Interrupt Request (IRQ), which is used with the clocks; the Fast Interrupt Request (FIRQ), which is used to detect the presence of a ROM-pack in the side slot; and the Non-Maskable Interrupt (NMI), which doesn't seem to be used at all here. In order to enable to IRQ, bit #4 of the Condition Code register must be cleared. This can be done with ANDCC #%11101111, leaving all other CC bits unchanged. If that same bit #4 were set to one (with ORCC #%00010000) then the IRQ is masked, or shut off.

When the IRQ is triggered by a clock pulse, the processor pushes all the registers onto the stack. Their values are saved because the interrupt service routine might alter some of them, and we'll want to restore the registers to their original values before returning to the main program. (The FIRQ only stacks the CC and PC registers, but that's another story.) The PC is then automatically loaded with what is in addresses $FFF8 and $FFF9. This is a ROM area and these contents can't be changed. The contents of $FFF8-9 are $0100, and so control passes to that address. At $0100 we find the instruction JMP $A9B3. But now we're in a RAM area, and as such we can change the destination of that instruction so that it jumps to the start of our own interrupt service routine. We'll get to that later. Remember that if the mask bit for the IRQ is set, then the above sequence will not take place.

The interrupt signal is provided from outside of the 6809E CPU, but it doesn't come in directly. It is instead routed through a PIA (Peripheral Interface Adaptor), which can be described as a chip which allows the CPU to communicate with the outside world. (For a more complete description of the 6821 PIA, see Lance Leventhal's "6809 Assembly Language Programming".) The next stage required in enabling a clock interrupt involves PIA#0 at $FF00-$FF03. There are two PIAs on the Color Computer (the other is at $FF20-$FF23), and each is composed of two ports. A separate clock interrupt signal is available on each of the two ports of PIA#0; one operates at 63.5u sec, the other at 60Hz. In turn, each port is composed of three separate registers-a Control Register, a Data Register and a Data Direction Register.

In PIA#0, the CR of Port B is accessed through address $FF03, and the Data Register is accessed through $FF02 (the DDR isn't used here). The 60 Hz clock signal is available on Control Line #1 of this port. To enable this interrupt, you simply set bit #0 of the Control Register. This is accomplished with:

```
LDA $FF03
ORA #1
STA $FF03
```

or, since the Control Register usually contains the number $34

```
LDA #35
STA $FF03
```

There is one remaining consideration. When an interrupt occurs on Control Line 1, bit #7 of the Control register goes high. This bit must be cleared before another interrupt can occur on that line. The only way to clear that bit is to read the Data Register on that same port, i.e., LDA $FF02. The interrupt is thus re-enabled for the next cycle.

We now have all of the ingredients necessary

to utilize a clock interrupt (see listing #1).

When the interrupt occurs (at 60 times per second), control will automatically pass to the instruction at $0100. This instruction is altered by the first two lines of code above so that it jumps to the INTSRV routine. In INTSRV, a location in video RAM was chosen as the clock counter so that you can see it being incremented on the screen. Notice that upon exiting the program, the clock stops counting even though the 60 Hz interrupt is still active. Both links in the chain must be enabled in order for the routine INTSRV to be called.

You can build any number of separate clocks, or use a series of ascending counters to provide larger units of time. For example, when the first clock reaches 60, you clear it and then increment a "seconds" clock. When that reaches 60, the "minutes" clock is incremented, and so on.

The 63.5 u sec interrupt which is available on Port A is much faster. Using it means that you'll spend a lot more time away from the main program, and the resulting slowdown might be very noticeable. This interrupt is implemented in the same way as the 60 Hz clock, except that addresses for Port A are used.The Control Register for Port A is at $FF01. Reading the Data Register at $FF00 might interfere with keyboard input, as described in Sept. Color Computer News.

While not practical, it is possible to use both clock interrupts at the same time. But since they both occur in IRQ, they result in a jump to the same service routine. It is therefore necessary to distinguish which interrupt did the calling. This can be accomplished by testing bit #7 of the appropriate Control Register, since that bit does go high when an interrupt occurs there.

The entire procedure is as shown in listing #2.

The clock interrupts as described allow the computer to share its time between the main program and the service routine. ON the other hand, use of the SYNC or CWAI instructions result in the processor just standing still until an interrupt occurs. You could use that approach when you have a block of code to performed at discreet intervals, and nothing else to do between those times. An example could be listing #3.

In this case the IRQ is not enabled. The processor waits at the SYNC instruction until an interrupt occurs; and then, instead of jumping to a service routine, the CPU picks up execution at the address following the SYNC.

```
*LISTING #1
BEGIN  LDX #INTSRV
       STX $010D  SET JUMP VECTOR
       LDA #$35
       STA $FF03  ENABLE 60HZ INT
       ANDCC #%11101111 IRQ ON
LOOP   JSR ($A000) TO POLCAT
       BEQ  LOOP    LOOP UNTIL ANY KEY
PRESSED
       ORCC #%00010000 IRQ OFF
       SWI    EXIT TO MONITOR

INTSRV INC  1234  CLOCK COUNTER
       LDA  $FF02 CLEAR STATUS BIT#7
       RTI

*LIST #2
BEGIN  LDX #INTSRV
       STX $010D SET VECTOR
       LDA #$35
       STA $FF01 63.5 uSEC ON
       STA $FF03 60 HZ ON
       ANDCC #%11101111 IRQ ON
LOOP   JSR ($A000)
       BEQ LOOP
       ORCC #%00010000 IRQ OFF
       LDA #$34
       STA $FF01 63.5uSEC OFF
       STA $FF03 60 HZ OFF
       SWI TO MONITOR
INTSRV LDA $FF01 CR FOR PORT A
       BMI INTPA 63.5 SEC IS ACTIVE
       INC CLK1  60 HZ CLOCK
       LDA $FF02 CLEAR BIT 7 PORT B
       RTI
INTPA  INC CLK2 63.5 SEC CLOCK
       LDA $FF00 CLEAR BIT 7 PORT A
       RTI

*LISTING #3
       LDA #$35
       STA #FF03 60 HZ INT ON
LOOP   LDA $FF02 CLEAR BIT 7
       SYNC  WAIT FOR INTERRUPT
       ..... YOUR PROGRAM HERE
       BRA LOOP FOR NEXT CYCLE
```

COMMENT CORNER

by Andrew Phelps
The Micro Works

The following is a list of comments which could be added to a disassembly listing of the Color Computer ROM. The subject this time is cassette I/O.

All data is stored on tape in blocks. Each block has a type and 0-255 bytes of data. The valid types are:

$00 - header block
$01 - data block
$FF - end of file block

Variables, areas, and routines -

Addr	Comments
007C	BLOCK TYPE
007D	BLOCK LENGTH
007E	BUFFER ADDRESS
0080	CHECKSUM
0081	BYTE COUNT & ERROR STATUS
0082	GENERAL COUNTER
0083	PULSE WIDTH COUNT
0084	RISE / FALL FLAG; 0=>RISE
0085	LAST SINE VALUE
008A	TWO BYTES ALWAYS ZERO
008F	FULLWAVE THRESHOLD (=12)
0090	NOISE REJECTION (=18)
0091	HALFWAVE THRESHOLD (=0A)
0092	GAP LENGTH (=0080)
A004	ADDRESS OF: READ GAP
A006	ADDRESS OF: READ BLOCK
A008	ADDRESS OF: WRITE BLOCK
A00C	ADDRESS OF: WRITE GAP
A6F3	BLINK CORNER OF SCREEN
A701	READ GAP AND BLOCK
A70B	READ BLOCK ROUTINE
A749	READ ONE BYTE
A755	READ ONE BIT
A77C	READ GAP ROUTINE
A7CA	MOTOR ON & DELAY
A7D8	WRITE GAP ROUTINE
A7F4	WRITE BLOCK ROUTINE
A82A	WRITE ONE BYTE
A85C	SINE TABLE
A880	LAST+1 SINE TABLE

Line-by-line comments -

Addr	Comments
A6F3	GET FROM UPPER-LEFT SCREEN
A6F6	SWITCH COLOR
A6F8	CHECK CURRENT LINE NUMBER
A6FA	WAS IT $FF?

Each block may be preceded by a gap. This gap allows the tape recorder to be started and stopped between records, both when writing the file and when reading it. This gap consists of 128 bytes of "01010101". There is always a gap before a header block to allow the read routine to properly sync to the incoming data. On a text file there is a gap before every block.

A6FB	SKIP IF NOT IMMEDIATE MODE
A6FD	STORE INVERTED CHARACTER
A700	RETURN
A701	CALL READ GAP
A703	CALL READ BLOCK
A705	ENABLE IRQ & MOTOR OFF
A708	GET ERROR STATUS TO B
A70A	RETURN
A70B	INHIBIT INTERRUPTS
A70D	BLINK CORNER OF SCREEN
A70F	GET BUFFER ADDRESS
A711	PREVENT FALSE SYNC
A712	GET A BIT FROM TAPE
A714	ROTATE INTO A REG
A715	IS IT SYNC CHARACTER? (3C)
A717	LOOP TIL IT IS
A719	GET BYTE FROM TAPE
A71B	STORE AS BLOCK TYPE
A71D	GET ANOTHER BYTE
A71F	STORE AS BLOCK LENGTH
A721	ADD TO BLOCK TYPE
A723	START CHECKSUM
A725	GET LENGTH AGAIN
A727	STORE IN BYTE COUNTER
A729	IF ZERO LENGTH, SKIP
A72B	GET A BYTE FROM TAPE
A72D	STORE INTO MEMORY
A72F	TEST MEMORY; INCREMENT X
A731	IF NO MEMORY, ERROR #2
A733	ADD ONTO CHECKSUM
A735	STORE CHECKSUM
A737	COUNT DOWN BYTE COUNTER
A739	LOOP TIL ALL BYTES READ
A73B	GET ONE MORE BYTE
A73D	EQUALS CHECKSUM?
A73F	GO CLEAR ERROR IF MATCHES
A741	ERROR #1 - CHECKSUM ERROR
A743	SKIP TWO BYTES
A744	ERROR #2 - NOT RAM
A746	STORE AS ERROR CODE
A748	RETURN
A749	EIGHT BITS PER BYTE
A74B	TO COUNTER
A74D	GET ONE BIT
A74F	ROTATE INTO A REG
A750	COUNT DOWN BIT COUNT

```
A752  LOOP FOR ALL BITS
A754  RETURN
A755  GET TIME TIL NEXT EDGE
A757  GET TIME COUNTER
A759  BACK UP ONE
A75A  COMPARE WITH THRESHOLD
A75C  RETURN WITH BIT IN CARRY
A75D  INITIALIZE TIME COUNTER
A75F  WHICH EDGE SHOULD LOOK FOR
A761  SKIP IF FALLING EDGE
A763  COUNT AND TEST INPUT
A765  LOOP UNTIL LOW
A767  COUNT AND TEST INPUT
A769  LOOP UNTIL HIGH
A76B  RETURN
A76C  BUMP TIME COUNTER
A76E  INPUT PIA DATA
A771  MOVE BIT INTO CARRY
A772  RETURN
A773  COUNT AND TEST INPUT
A775  LOOP UNTIL HIGH
A777  COUNT AND TEST INPUT
A779  LOOP UNTIL LOW
A77B  RETURN
A77C  INHIBIT INTERRUPTS
A77E  MOTOR ON AND DELAY
A780  CLEAR COUNTER
A782  FIND RISING EDGE
A784  COUNT TIL LOW
A786  IF WIDE, TRY RISING SYNC
A788  COUNT TIL HIGH
A78A  IF NARROW, TRY RISING SYNC
A78C  COUNT TOWARD FALLING SYNC
A78E  GET SYNC TYPE COUNT
A790  IF 96 OF SAME TYPE, OK
A792  IF NOT, GET ANOTHER SYNC
A794  SET RISE/FALL FLAG, 0=RISE
A796  RETURN
A797  COUNT TIL HIGH
A799  IF WIDE, BACK TO FALL CITY
A79B  COUNT TIL LOW
A79D  IF NARROW, TRY FALL AGAIN
A79F  COUNT TOWARD RISING SYNC
A7A1  GET SYNC TYPE COUNT
A7A3  96 OF THIS TYPE?
A7A5  GO TEST; LOOP OR LEAVE
A7A7  INIT PULSE WIDTH COUNT
A7A9  COUNT TIL HIGH
A7AB  GO CHECK FOR NOISE
A7AD  INIT PULSE WIDTH COUNT
A7AF  COUNT TIL LOW
A7B1  GET PULSE WIDTH
A7B3  WAY TOO WIDE?
A7B5  FORGET IT IF IT IS
A7B7  CHECK SYNC THRESHOLD
A7B9  RETURN WITH CARRY IF WIDE
A7BA  NOISE; START <<ALL>> OVER
A7BC  RETURN

A7CA  GET PIA CONTROL REGISTER
A7CD  SET MOTOR ON BIT
A7CF  STORE BACK TO PIA
A7D1  GET ZERO TO X
A7D3  COUNT DOWN
A7D5  LOOP FOR 0.6 SECONDS
A7D7  RETURN
A7D8  INHIBIT INTERRUPTS
A7DA  MOTOR ON AND DELAY
A7DC  GET GAP LENGTH
A7DE  WRITE ONE BYTE: 01010101
A7E0  COUNT DOWN X
A7E2  LOOP FOR GAP LENGTH
A7E4  RETURN
A7E5  MOTOR ON; WRITE GAP
A7E7  WRITE BLOCK
A7E9  ENABLE INTERRUPTS
A7EB  GET PIA CONTROL REGISTER
A7EE  MOTOR BIT OFF
A7F0  WRITE BACK TO PIA
A7F3  RETURN
A7F4  INHIBIT INTERRUPTS
A7F6  GET BLOCK LENGTH
A7F8  PUT TO BYTE COUNTER
A7FA  GET LENGTH AGAIN
A7FC  IF ZERO LENGTH, SKIP
A7FE  GET BUFFER ADDRESS
A800  ADD BYTE ONTO CHECKSUM
A802  COUNT DOWN LENGTH
A803  LOOP FOR CHECKSUM
A805  ADD ON BLOCK TYPE BYTE
A807  STORE TO CHECKSUM
A809  GET BUFFER ADDRESS AGAIN
A80B  WRITE BYTE: 01010101
A80D  GET START MARK: 00111100
A80F  WRITE BYTE TO TAPE
A811  GET BLOCK TYPE
A813  WRITE TO TAPE
A815  GET BLOCK LENGTH
A817  WRITE TO TAPE
A819  LENGTH = 0?
A81A  IF ZERO LENGTH, SKIP
A81C  GET DATA BYTE
A81E  WRITE IT TO TAPE
A820  COUNT DOWN BYTE COUNTER
A822  LOOP TIL ALL DATA WRITTEN
A824  GET CHECKSUM
A826  WRITE IT TO TAPE
A828  GET: 01010101
A82A  SAVE OUTPUT BYTE
A82C  INITIALIZE BIT COUNT FLAG
A82E  GET END OF LAST WAVE
A830  OUTPUT TO DAC
A833  GET ADDRESS OF SINE TABLE
A837  GET BIT TO OUTPUT
A839  SKIP IF A "1"
A83B  GET SINE BYTE, Y=Y+1
A83D  OFF END OF TABLE?
A841  GO LEAVE IF SO
A843  PUT BYTE TO DAC
A846  LOOP FOR REST OF WAVE
A848  GET EVERY OTHER BYTE; Y=Y+2
A84A  OFF END OF TABLE?
```

```
A84E  GO LEAVE IF SO
A850  PUT BYTE TO DAC
A853  LOOP FOR REST OF WAVE
A855  SAVE LAST BYTE OF WAVE
A857  MOVE BIT FLAG OVER
A858  LOOP FOR NEXT BIT
A85A  RESTORE AND RETURN
A85C  START OF SINE TABLE
A880  END OF SINE TABLE + 1
```

QUESTION: What do I need to do in order to write a block to tape?

Put the address of the data in $7E, the block type in $7D, and the number of bytes (0-255) in $7C. Then call a routine to write the block.

Which write routine should be called?

If you're writing a data block or end-of-file block in a binary file, just call "WRITE BLOCK". For a header record or an Ascii file, call "WRITE GAP" and then "WRITE BLOCK".

How is a block read in?

Put the address of your buffer into location $7E, and call the routine. The length of the block read and its type will be set in $7D and $7C. Location $81 will be zero if there was no error.

Which read routine should be called?

If the tape recorder is going and no gap is expected, then call "READ BLOCK". Otherwise, call "READ GAP AND BLOCK".

What is the purpose of the gap?

When the gap is read back in, the computer can tell when the motor is up to speed and the data is valid. It can then determine whether it should be looking for negative or positive edges to mark the end of bits. This must be done at least once; it is done before every record in an Ascii file because the tape recorder is turned off in between each record in Ascii files.

How is a bit recorded?

It is recorded as a wave, since waves are what tape recorders are designed to record. A "1" is recorded as a wave which takes 500 microseconds; a "0" as a wave which takes twice that long.

What is this "wave" as it leaves the computer?

It is a voltage which increases, then decreases down past its original value, then increases back up to its original value. This voltage is generated under program control, using the table at location $A85C.

How does the computer generate a voltage?

A number corresponding to the desired voltage is stored in location $FF20, which is hooked to a "DAC", or "Digital to Analog Converter".

How is a bit read back in?

Bit zero of location $FF20 is an input which goes back and forth between one and zero as each incoming wave goes by. By counting the time between changes of this bit, it can be determined whether the wave was a long one ("0") or short one ("1").

What about the section of the read gap routine which keeps fussing about "rising" versus "falling"?

The tape recorder could possibly send the waves back upside-down.

That is, the voltage would fall, then rise, then fall, instead of the other way around. If this is so, the read gap routine can figure this out and set a byte ($84) so that the read block routine will look for "low-high" as one wave instead of looking for "high-low".

What is the Baud rate of the tape?

The Baud rate generally means how many bits are recorded each second. In this cassette recording scheme, there is no fixed number for this since some bits are longer than others. A file of all ones would record at 2000 Baud, but a file of all zeroes would record at 1000 Baud. A typical file of half ones, half zeroes would record at 1333 Baud (NOT 1500 Baud; think about it.)

Why are interrupts inhibited while reading or writing the tape?

All timing is done by the program. If the write routine were interrupted, there would be a wave which paused in the middle. If the read routine were interrupted, it might miss a wave and could not accurately determine how long each wave is.

What causes I/O errors?

Besides the obvious causes such as dirty or wrinkled tape, errors can be caused by operator procedure. For example, if you fast-forward past the leader at the beginning of the tape, you could go past an old header block on the tape and end up making a file with two header blocks; this would cause an I/O error when you tried to read it. Generally, however, the cassette system on the Color Computer is not prone to errors and is efficient and reliable.

Sam Mix Software

3424 College N.E.
Grand Rapids, MI 49505

THREE GAME PACK

BATTLESHIP, OTHELLO, LIFE

$19.95

CONNECT FOUR

Challenging game played either with two players or against the computer.4K

$12.95

CCN SAMPLERS

1-2-3. A collection of all programs appearing in CCN plus more.

$7.95 each

MOON LANDER

Just Arrived!

Reviewers say,
"Just like the arcade games;
just like flying."

Requires 16K extended basic.

ATTENTION HAMS!!!

CW Software — $19.95
with appropriate interface use your CC to send and receive CW.

MFJ 1200 — $69.95
CW Computer Interface System
6-9 AC supply . . . **$9.95**

Add $1.00 Postage & Handling.

★ *LOOKING FOR NEW SOFTWARE* ★
— TOP ROYALTIES PAID —

Money Order's sent same day. Personal checks must clear your bank before shipment.
Michigan residents add 4% Sales Tax.

Star Blaster

Maria Crowder

I don't usually let my own staff write reviews but Marie wrote such a dynamite review of Star Blaster that I just couldn't refuse. Marie is CCN's newest employee and, as you'll see, one very sharp kid. Her tips for winning really work. Bill

If you're looking for a highly-technological, complicated review, read no further. This writer is an ordinary sixteen year old. Ordinary except for one thing. Instead of going to movies, bowling, and school activities like most teen-agers, I spend almost all of my free time (and money) at nearby video arcades. I had to get a job at a booming computer firm because my new hobby was costing my parents about $10 worth of quarters every week. My dad would probably have to morgage our house if I didn't! Now, about half of my paycheck goes into the quarter slots of "Pac Man", "Defender", "Galaxian", and "Space Invaders". But out of all the flashing, fast-paced games, I would have to say that my all-time favorite game is "Asteroids" by Bally. You know, the game where you're a ship in the middle of an unknown universe with rocks and strange UFO's flying at you in all directions and there's no escape. When you think you've defeated the enemies and wiped out every little rock and pebble, the universe once again fills with asteroids and UFO's, but this time more than before. It's a no-win situation. Then, when your last ship is blown up by a UFO, the thrill is gone and life once again becomes a reality.

The reason for all of this ridiculousness is to review a new game by The Micro Works called Starblaster for the Radio Shack 16K Color Computer. This game is a combination of Asteroids and Asteroids Deluxe. It is like Asteroids Deluxe because it uses a shield to defend your ship instead of a hyperspace. The way the game works is fairly simple. When you blow up a large rock, it turns into two medium rocks; and when you hit a medium rock, it turns into two smaller rocks. Easy, huh?

The point system is as follows:
large asteroid = 10 points, medium asteroid = 50 points small asteroid = 100 points large ship = 500 points small ship = 1000 points

To begin with, each player gets three ships and three force shields. The force shields can be used at any time during the game to ward off attacks but remember you only have three to begin with. After every 10,000 points, though, you get one free ship and three force shields.

The keys used in the game are:
"D" rotates left, F rotates right, left arrow fires, right arrow controls thrust and the space bar controls the force field (Half the battle is learning to maneuver your ship so practice, practice, practice!).

I feel that "Star Blaster" is a little easier than "Asteroids", but it is still a challenge to amateurs and pros alike. One possible reason that "Star Blaster" is a little easier could be in the sound. In Asteroids, a pounding noise simular to your own heartbeat is heard throughout the game. As the game progresses, the beeps get faster and faster, along with your heartbeat. Yes, psychology is used in video games too! "Star Blaster" doesn't have this feature. Also, you can practice "Star Blaster" more often because it isn't a quarter for every game.

The disadvantages in this game, I feel, are: 1) This game doesn't take advantage of the color on the Radio Shack Color Computer. But, Bally's "Asteroids" is in black and white and "Star Blaster" at least is in black and green and 2) There is no top ten players list like "Asteroids" in which I look forward to putting my initials in.

Now, let's get down to the basics. Do you want to learn how to be the next U.S. "Asteroids" champ in Washington D.C? Here are a few ground rules:

1. The rocks always form on the sides of the screen, so stay in the middle of the screen when a new screen begins.
2. Smaller rocks are harder to shoot and usually move faster so it is best to shoot at little rocks before big rocks.
3. It is sometimes profitable to leave only one small rock on the screen and shoot the UFO's when they appear since they're worth so many points.
4. The more times you clear the screen of rocks, the more big rocks appear on the screen (up to sixteen) so be prepared.

So, overall, I truly recommend this game to anyone with a Radio Shack Color. I think you'll enjoy this one and it is really worth your money.

SDS80C
by Mark Rothstein

The Micro Work's Software Development System is a ROM-based program for the Color Computer which allows a user to do the three computer related tasks necessary to build a working assembly language program. It consists of three distinct programs an editor, an assembler, and a debugger. In this article I will review how the development system is used, each of the three programs that make up the development system, then briefly, the owner's manual.

Overview--How SDS80C is used.

Once a programmer has decided what he wants a program to do, he then decides how to implement it--he dedided on the logical flow of the operations. He must then translate these operations into computer instructions. Using the Editor, a programmer can use his Color Computer as a word processor to enter and edit these instructions; the result is called source code. It includes instruction like ADD A, STA, CMP X (all in the standard Motorola format) and to insure the programmer's sanity two months from now, lots of comments. When editing is finished, the programmer can invoke the Assembler. It converts the source code into two outputs: a listing output, and object code output. The listing tells the programmer what he has instructed the computer to do. The object code is the program in computer executable form. At the programmer's option, this object code will be stored in memory where he can execute it via the ABUG monitor. This will allow him to test his code in a very thorough manner, segment by segment. The manual describes an easy way to test the code by using the SWI instruction. SWI puts ABUG back in control. Now when the programmer finds that the program is not doing what he expects it to, he can immediately return to the Editor, change his source code, and reassemble, all in a matter of seconds! I can't emphasize the efficiency of this approach enough. I like to see the unbiased results of my efforts. I guess that's why I'm an engineer. This ability to thoroughly check out a program and to isolate a problem and fix it quickly really gives me a thrill. I can say, once you develop programs this way, you won't willingly do it any other way.

The Editor.

When the SDS80C is fist turned on, it starts in the edit mode. One or more source code files can then be loaded from the cassette interface, or the source code text can be keyed in. The editor is screen-oriented; Only the text that appears on the screen when a user scrolls through the text buffer will be assembled or stored on tape at the user's command. As in BASIC, there is a blinking cursor which indicates where you are in the buffer at any moment. Another very nice feature is that the editor continuously indicates how much memory is available.

The Editor is very flexible and allows a user to:

Key in text (at the cursor position).

Read from the cassette interface and place the text in the buffer, or append text to the buffer contents.

Scroll through the buffer one charactor or a line at a time (and repeat at a comfortably fast rate).

Duplicate a block of text.

Move a block of text from one location to any other area in the text buffer.

Jump to the beginning or the end of the text buffer.

Find a string.

Change a string.

Repeat the find and change commands.

and more.

These commands are adequate to conveniently manipulate a source code file.

To get you started Micro Works provides an innocent looking sample program which is an excellent way for a user to begin using SDS80C. This program can be easily keyed in (it's only ten lines long) and assembled. Then each time it is executed, it inverts the charactors on the screen. (It changes the text from light charactors on a green background to green characters on a light background, and vice versa). In my estimation, this is a good example of giving a user immediate positive feedback on how easy it is to edit a program (and assemble and test it).

The only deficiency I found in the editor is that the maximum length of a line is 32 characters. Sometimes I found my comments running onto an undesired new line. This would convert the assembly editor into a generalized text editor. The difficulty of implementing this capability, however, is formidable and also involves changes to the assembler. With this in mind, Micro Works has made a good compromise.

As a final note on the Editor, one very interesting feature is its ability to recover a text file after a user program has "crashed". In executing this command the Editor starts from the beginning of its memory and puts into the buffer anything that looks like text. What a lifesaver! Usually the text file remains intact and garbage appears only at the end of the file. Don't make a habit of this, though; sometimes the text file is all garbage.

Exiting from the Editor, a user may either go to

the Assembler by two keystrokes (@ enter) or to the monitor via three.

The Assembler.

The key features of the Assembler are that it supports both the Motorola 6809 and the 6800 instruction sets; it permits local and global labels; it supports conditional assembly commands; it has a number of assembly listing options; it generates error messages, even when the listing is suppressed; and it can send object code output to memory or the cassette interface or both.

The only feature I expected in a product of this quality, but didn't find, was a macro capability. This would give the programmer the ability to write a section of code, like a subroutine; however, when it assembled it would generate in line code. See one of the Assembly language books for additional discussion.

The first key feature is the full 6800 and 6809 instruction sets. The reason for supporting the 6800 instruction set is to provide upward compatibility from the 6800 to the 6809. This means that a program written for the 6800 can be easily modified to run on the 6809; sometimes this only involves reassembly of the source code. Another reason for including the 6800 instructions is that some of them are clearer than their 6809 counterparts; for instance ABA, CLC, INX, etc.

The second key feature is local labels. Global labels are a standard feature of assemblers; they appear in the symbol table and can be referenced from any part of the source code. Local labels, however, are a convenience; they are symbols which are defined only within a small section of code.

The conditional assembly, the third feature, allows two or more versions of a program to be generated from the same source code. This is useful when a program must be written and maintained for computers with different I/O, for a stripped-down version, etc.

The fourth feature, the listing options, includes assembly output with or without a symbol table; output to the screen or to a 32, 40, or 80-column printer; and a choice of three speeds (fast, slow, and pause). Error message are output even when the listing is disabled.

An optional feature is Generating object code. This is useful when the programmer wants to check for syntax errors and undefined symbols. When generated and placed in memory, the object code will be stored right after the source code unless the programmer specifies otherwise. During assembly, the symbols and their addresses are also saved for later use.

Exit to the monitor is automatic.

The Monitor.

The ABUG Monitor allows a user to conveniently check out his program. Via a single letter entry, the user can do any of the following:

Examine and Change Memory
Examine and Change registers
Evaluate expressions
Move a block of memory
Free up the stack
Jump to user-specified address
Return to the Editor
Save memory on cassette
Execute the Object Program
Load from cassette

I will describe the first four commands, the most interesting ones.

The first command involves a procedure for examining and changing memory data, and it is one of the more user-convenient ways of looking into memory that I've seen! On entering the command "M" and a four digit hexadecimal address, the monitor displays the contents of not one, but eight locations in memory, stating at the next lower 8-byte boundary. A user can then continuously walk his way thru memory using the four arrow keys, and changing data when he pleases.

The second command feature is the expression processor. As a distinct command, it allows a user to do simple arithmetic in decimal and hexadecimal and to get the value of a label or symbol defined in the assembly process. (Incidentally the monitor shares this processor with the assembler). Unfortunately the expression evaluator cannot be used directly with the examine and change memory command.

Occasionally, a program will occupy most of available memory, making execution of the program impossible. A third command, "Unstack", removes all symbols and their values from the stack and frees up this area. The symbols will then not be available for the expression evaluator.

A fourth command, by a single keystroke, allows the user to return to the editor for another try at the program. If, however, the computer has "gotten lost in the weeds", the only way to get its attention may be to push the little black button on the back-reset. This will cause SDS80C to restart with the Editor. However, you will recall that Editor has a command which will attempt to ressurect the source text.

Outputs, etc.

Output normally is sent to the TV screen, but can be switched to the device on the RS-232 port instead. Standard printers which run with BASIC will run with SDS80C. However, if you've read my other articles, you know that I use a terminal

continued on page 30

Eighty/Apple Computer Show, NY Statler Hotel, New York, NY. The Second Annual Eighty/Apple Show is an exposition of products and services oriented to the TRS-80 and Apple Computer Systems. The show will feature over 100 exhibitors of hardware, software, books, magazines, supplies, services and accessories for these two popular small business and home computer systems.
FOR ADDITIONAL INFORMATION CONTACT:
Ken Gordon
Kengore Corporation
3001 Route 27
Franklin Park, NJ 08823
(201)297-2526

New and Custom software for the TRS-80 color computer.
Write to:
Soft Data
P.O. Box 80019
Albuquerque, New Mexico 87108

The Capital Children's Museum of Washington, D.C. and Reston Publishing Company (A Prentice-Hall Company) are pleased to announce an agreement to produce a series of educational book/software packages for microcomputers. The series is designed to promote a creative, interactive use of computers by children, parents, and teachers. The product will be implemented initially on the Atari 800 Personal Computer System; design and testing are being managed by Superboots, the software development arm of the Capital Children's Museum.
The first package, entitled Paint, will be available in early 1982. Paint will be a versatile educational tool that will not be suitable for use either in the home or in a classroom setting. The book accompanying the software will be a guide to a wide range of activities which parents or teachers can use to extend a child's interest in computer learning.
Early in 1981, Atari made a significant donation to the Capital Children's Museum in the form of 30 Atari microcomputer systems. This contribution has allowed the Museum to establish a computer learning enviroment called FUTURE CENTER, to put computer programs in exhibits, and to utilize other computers in Superboots. Superboots is the software development lab where computer programs are created. Software is used both in FUTURE CENTER and in exhibits and is marketed outside the Museum through Reston Publishing Company.

For more information, contact:
Bob Evans, Administrator, Superboots,
Capital Children's Museum, 800 Third Street, N.E., Washington, D.C. 20002 (202) 543-8600
Nikki Hardin, Editor,
Reston Publishing Company,
11480 Sunset Hills Road,
Reston, VA 22090 (703) 437-8900

MFJ introduces a CW transceive program complete with hardware interface for the Radio Shack TRS-80 Model I and Model III computers.
It lets you send CW on your keyboard and receive CW on your display screen and features a tri-split screen for received messages, transmit buffer, and programmable message index.
A huge 3295 character (with a 16 K machine) text buffer makes sending CW effortless even if you "hunt and peck".
You can preload the text buffer while receiving and then transmit when ready. There are ten 199 character programmable message memories with an on-screen message index. You can repeat and combine these messages together as needed. Speed is adjustable from 12 to 55 WPM. For group code practice you can store up to 2200 characters.
On receive, there is an exclusive keyboard adjustable "Fist Fixer" with an on-screen analog indicator. This alters the computer decision making process and improves copy of poorly sent CW. The program automatically receives up to 100 WPM and lets you store up to 5 screen full of received CW for logging at a more convenient time. When the transmit buffer is empty the mode changes automatically back to receive.
The hardware interface plugs between the transceiver and computer. No modification is needed to the rig or computer and nothing else is needed.
Optimized RTTY techniques are used in the hardware interface. It features an automatic noise limiter, a narrow 4 pole active bandpass filter, an active lowpass post detection filter, and a tracking comparator.
It has high voltage keying circuits that will key virtually any tube or solid state transmitter. There are LEDs that aid in tuning, indicate a transmit condition and a power "ON" condition.
The interface requires 9-18 VDC or 110 VAC with optional MFJ-1312 AC adapter, $9.95. An all aluminum cabinet is used for RF shielding and measures 6 x 1 3/4 x 3 inches.
The program and interface requires the Radio Shack TRS-80 Model I or Model III with at least 16 K of RAM.

The program is supplied on cassette tape and is disk compatible.
MFJ provides a 30 day money back trial period. If you are not satisfied, you may return it within 30 days for a full refund (less shipping). MFJ also provides a one year unconditional hardware guarantee.
The MFJ-1210 (for the Model I) and the MFJ-1212 (for the Model III) is available from MFJ Enterprises, Inc. for $99.95 plus $4.00 shipping and handling for each unit.
To order, call toll-free 800-647-1800 and charge Visa or Master Charge or mail order to:
MFJ Enterprises, Inc.
P.O. Box 494
Mississippi State, MS 39762.

Just announced from MFJ Inc. a new Inductive Coupled Modem. Eliminates room noise, vibration and other acoustic coupled problems. Half/full duplex, crystal controlled. RS-232, TTL, CMOS. Bell 103 compatible. What makes this modem different?
(First) It uses inductive coupling for receiving. This innovative technique eliminates room noise, vibration and other acoustic coupled problems. The result is more reliable data transfer. (Second) It is RS-232 compatible and provides TTL and CMOS input/outputs. Lets you interface to nearly any computer with proper software. (Third) Cassette recorder input/output jacks let you record your transmitted data and received data and load it back to your computer or retransmit it later. (Fourth) It has Originated/Answer modes and Half/full duplex operation. (Fifth) It is crystal controlled for high stability. (Sixth) It has low price and excellent quality. (Bell 103 compatible) Carrier detect, power "ON" LEDs. 0 to 300 baud. All aluminum cabinet. Simple to install and operate. Made in USA.
MFJ-1230--$129.95.
For additional information write: Tom Mix Software
3424 College N.E.
Grand Rapids, MI 49505

The Index, is a unique information resource. It compiles over 12,000 articles, editorials and columns from over 900 issues of personal computer magazines written in the last six years into over 30,000 index entries.
Its purpose is to index by subject the massive amount of information that has always been available to hobbyists andbusiness users, but which has been practically impossible to gather and maintain.
From the beginning this ambitious job of first gathering the magazines, then sorting out the 30,000 entries and then finally printing The Index has been undertaken by William H. Wallace.
Wallace, an attorney, is no stranger to Micro Computing. As a former employee of IBM, he's been an enthusiast for years.
"The more projects I researched and the more computing jobs I started, the more I realized how much time it took to find out what has already been done and what hasn't," Wallace said.
"I was getting five or six magazines already and could go to the library for a couple of more, but the hard part was always getting started; finding articles on the appropriate subject or finding out there hadn't been anything written on what I wanted to do."
Wallace decided it was time to compile The Index. But instead of just listing magazine articles by a limited number of subjects, he utilized a method of indexing called "Key Work in Context" (KWIC).
"KWIC was just a way to make my Apple Computer work for me," Wallace says. "The idea is that an article title has four or five words that can be key words for a specific subject. So, we just let the computer order the title in all the ways it could be used for subject indexing."
Wallace's Index not only indexes general articles, but indexes articles according to which machines that can be applied. The Index has separate indexes for Apple, Atari, TRS-80, and ten(10) more, plus a general index of over 5000 entries.
"Its really been fun and I'm very happy with the results, but now it's time to start work on adding more magazines and updating those we have."
For more information and to order The Index contact:
Missouri Indexing, Inc.
P.O. Box 301
St. Ann, MO 63074, 314/997-6470.)

NJ MICROCOMPUTER SHOW & FLEAMARKET, Holiday Inn (North) at the North Terminal of Newark Int'l Airport, Newark, NJ. This is a special 1-day Spring event (third annual), featuring over 50 commercial exhibitors and 150 fleamarket sellers. The regular two-day Fall Show will be held on September 11-12, 1982. The Computer Fleamarket will be held indoors in case of rain. Featured will be hardware, software and accessories for all popular systems including Apple, TRS-80, Atari, Pet, Heath/Zenith, ZX-80/81, S-100, IBM and others. For additional

information contact:
Ken Gordon, Producer
Kenqore Corporation
3001 Route 27
Franklin Park, NJ 08823
(201) 297-2526

This summer, youngsters can sign up for an overnight camp in Simsbury, Connecticut, where the main activity will be .. COMPUTERS. A second location is in Atlanta, Georgia. This unique recreational and educational experience is directed by Dr. Michael Zabinski, Professor at Fairfield University. Now in its fifth year, it is the original computer camp currently offered in the USA.
Four action-packed weeks are planned from July 11 to August 16. The campers, ages 10-18, will enjoy small group instruction and mini and micro computers for ample "hands-on". Dr. Zabinski will be assisted by elementary and secondary school teachers.
The camp is for kids of all levels of experience including no experience whatsoever. In addition to computers, the campers will enjoy superb recreational facilities including swimming and tennis.
For further information contact Michael Zabinski, Ph.D., at (203) 795-3049, or write:
National Computer Camp
P.O. Box 624
Orange, CT 06477

A new catalog from Electric Specialists presents their line of Microcomputer interference control products.
Protective devices are also included.
Descriptive sections are included which outline particular problems. Suggested solutions are given. Typical applications are uses are also outlined. Request Catalog 811.
Electronic Specialist, Inc.
171 South Main Street
Natick, MA 01760 Phone: (617) 655-1532

The Eighty System Newsletter is a weekly newsletter for personnel who are users, retailers, distributors, or manufacturers of TRS-80 System related products. Mailed every Friday by First Class Mail, the newsletter provides up-to-the-minute news on what's happening in the TRS-80 industry. Also included each week is a bibliography of articles published in over 100 publications that relate to the TRS-80 System. For details or subscriptions ($39 per year-sample copy $3.00) contact:
Ken Gordon, Publisher
Kenqore Corporation
3001 Route 27
Franklin Park, NJ 08823
(201) 297-2526

Computerware has introduced a new board that expands the memory of the Radio Shack Color Computer from 16K to 32K.
Computerware's 16 PLUS BOARD does NOT require soldering, plugs in easily, and fits neatly under the RF shield cover inside the computer. Complete installation instructions are included with the board.
Computerware's exclusive design allows the graphics display to reside anywhere within the 32K of memory. No software modifications are required for exisiting software and the 16 PLUS makes your Color Computer completely compatible with the anticipated disk systems.
The 16 PLUS BOARD costs only $84.95 plus $2.00 S/H and is available directly from Computerware at Box 668, 1472 Encinitas Blvd., Encinitas, Ca. 92024, (714)-436-3512.

ELECTRONIC SPECIALISTS expands their AC Power Line Interrupter series to include automatic reset models. Should AC Line Voltage be disrupted or exceed pre-set safety limits, the POWER INTERRUPTER disconnects AC power from controlled apparatus. A 4-minute time delay, followed by automatic self-reset, helps avoid wide voltage fluctuations associated with Power Line malfunctions. An optional Line Voltage Monitor is available.Intended for installations operation unattended for long periods, the SELF-RESET POWER INTERRUPTER can accommodate a 15 amp resistive load or a 10 amp inductive load.
Model PI-SR-15 Self-Reset Interrupter $185.95
Model PI-SRV-15 Self-Reset & Voltage Monitor Interrupter $205.95
Electronic Specialists, Inc.
171 South Main Street
P.O. Box 389
Natick, MA 01760 Phone: (617) 655-1532

Color computer owners,

32K PLUS DISKS*

$298.00

Yes, that's right - for as little as $298.00 you can add 32K of dynamic RAM, and a disk interface, to your TRS-80 Color Computer! If you just want the extra memory it's only $199.00, and you can add the disk interface later for $99.00.

Just plug the *Color Computer Interface (CCI)*, from Exatron, into your expansion socket and "Hey Presto!" - an extra 32K of memory. No modifications are needed to your computer, so you don't void your Radio Shack warranty, and Exatron give both a 30 day money-back guarantee and full 1 year repair warranty on their interface.

The *CCI* also contains a 2K machine-language monitor, with which you can examine (and change) memory, set break-points, set memory to a constant and block-move memory.

So what about the *CCI Disk Card*? Well as we said it's only an extra $99.00, but you'll probably want Exatron's *CCDOS* which is only $29.95 - unless you want to write your own operating system. The *CCI Disk Card* uses normal TRS-80 Model I type disk drives, and *CCDOS* will even load Model I TRSDOS disks into your color computer – so you can adapt existing TRS-80 BASIC programs.

As a further plus, with the optional *ROM Backup* adaptor, you can dump game cartridges to cassette or disk. Once the ROM cartridge is on cassette, or disk, you can reload, examine and modify the software. The *ROM Backup* adaptor is only $19.95.

For more information, or to place an order, phone Exatron on their Hot Line *800-538 8559* (inside California 408-737 7111), or clip the coupon.

excellence in electronics

exatron

DEALER ENQUIRIES INVITED

Exatron,
181 Commercial Street,
Sunnyvale, CA 94086

- ☐ Please send a 32K Color Computer Interface for $199.00
- ☐ Please send a CCI Disk Card for $99.00
- ☐ Please include CCDOS and manual for $29.95
- ☐ Also include a ROM Backup adaptor for $19.95

Please add $5.00 for shipping to all orders, and 6 percent sales tax in California.

Name .....

Address .....

City .....

State ..... Zip .....

Charge my:

☐ MasterCard Interbank Code .....

☐ Visa Expiration Date .....

Card .....

☐ Check enclosed for .....

☐ Ship COD ($2.00 extra)

Signature .....

*32K RAM plus Disk Interface TRS-80 is a trademark of Tandy